

R. Steinbrecher

Bildverarbeitung

in der Praxis

© Dr. R. Steinbecher, 2005

www.RST-Software.de

Das Werk darf frei nur unverändert und als Ganzes weitergegeben werden. Auszüge sind nur unter Angabe der entsprechenden Quelle (Zitat) zu verwenden. Die private Nutzung der aufgeführten Algorithmen ist ohne Einschränkungen erlaubt. Bei gewerblicher Nutzung ist zuvor das schriftliche Einverständnis der Firma Dr. R. Steinbrecher einzuholen.

Originaltitel in Buchform:

Steinbrecher, Rainer
Bildverarbeitung in der Praxis
München; Wien: Oldenbourg, 1993
ISBN 3-486-22372-0

Inhaltsverzeichnis

Vorwort	9
1. Einführung	11
2. Historie	15
3. Das menschliche Bildverarbeitungssystem	17
3.1 Der Aufbau des Auges	17
3.2 Die lichtempfindlichen Teile des Auges	18
4. Geräte zur Bildaufnahme	25
4.1 Die Kameraröhre	25
4.2 Die CCD-Kamera	27
4.3 Der Scanner	29
4.4 Fehler bei der Bildaufnahme	30
5. Geräte zur Bildwiedergabe	37
5.1 Die Kathodenstrahlröhre	37
5.2 Das LC-Display	43
5.3 Das Plasma-Display	44
5.4 Sonstige Wiedergabegeräte	46
5.5 Fehler bei der Bildwiedergabe und deren Beseitigung	47
6. Digitale Halbtonverfahren	55
6.1 Verfahren mit festem Schwellwert	55
6.2 Verfahren mit variablem Schwellwert	60
6.3 Grautonmuster	64
6.4 Kombinierte Verfahren	67
6.5 Dot-Diffusion	67
6.6 Digitale Halbtonverfahren bei Farbbildern	68
7. Bildrestaurierung	73
7.1 Beispiele für Störungen	73
7.2 Signalmodell zur Bilddegradation	73
7.3 Inverse Filterung	76
7.4 Constrained-Filter	77
7.5 Wiener-Filter	78
7.6 Interaktive Bildrestaurierung	79
7.7 Ortsvariante Bildrestaurierung	79

8.	Fourier-Transformation	81
8.1	Diskrete Fourier-Transformation (DFT)	85
8.2	Eigenschaften der Fourier-Transformation	86
8.3	Fast-Fourier-Transformation (FFT)	91
9.	Bildtransformationen	101
9.1	Diskrete Fourier-Transformation	101
9.2	Diskrete Sinus/Kosinus-Transformation	103
9.3	Walsh-Transformation	104
9.4	Hadamard-Transformation	105
9.5	Hauptachsen-Transformation	108
9.6	Weitere Bildtransformationen	111
10.	Bildverbesserung	113
10.1	Grauwertmanipulation	113
10.2	Grauwertquantisierung	116
10.3	Histogrammdarstellung	119
10.4	Medianfilter	126
10.5	Tiefpaßfilter	130
10.6	Hochpaßfilter	131
10.7	Butterworth-Filter	132
10.8	Kantenverstärkung	132
11.	Kantendetektion	137
11.1	Was ist eine Kante? Kantenmodelle	137
11.2	Anforderungen und Schwierigkeiten bei der Kantendetektion	139
11.2.1	Bewertung von Kantendetektoren	141
11.3	Einteilung der Kantendetektoren	142
11.4	Differenzoperator erster Ordnung	144
11.4.1	Form der Filtermasken	145
11.5	Differenzoperator mit einfacher Mittelwertbildung	148
11.6	Differenzoperator zweiter Ordnung	150
11.7	Template-Matching	152
11.8	"Optimale" Operatoren	154
11.9	3D-Operatoren	166
11.10	Kanten-Nachbearbeitung	167
12.	Skelettierung	175
12.1	Ein einfacher Algorithmus	176
12.2	Verfahren mit 3×3-Masken	177
12.3	Kontur-Folge-Verfahren	188
13.	Vektorisierung	191
13.1	Der Freeman-Kettenkode	191
13.2	Direkte Vektorisierung	193

14. Bildsegmentierung	203
14.1 Schwellwertbildung zur Bildsegmentierung	203
14.2 Segmentierung über Templates	205
14.3 Segmentierung mit Hilfe der Kantendetektion	206
14.4 Segmentierung durch Gebietswachstum	207
14.5 Morphologische Operatoren	209
14.6 Split-and-Merge	212
15. Bildkodierung	215
15.1 Kodierung von Binärbildern	215
15.2 Huffman-Kodierung	216
15.3 Lauflängen-Kodierung	219
15.4 Quadtree-Kodierung	219
15.5 LZW-Kodierung	220
15.6 JPEG-Verfahren	222
15.7 MPEG-Verfahren	225
16. Mustererkennung	229
16.1 Merkmale – Einfache Merkmale	229
16.2 Hough-Transformation	233
16.3 Fourier-Deskriptoren	241
16.4 Texturanalyse	243

A. Definitionen	251
B. Die VGA-Karte	261
B.1 Der technische Aufbau	261
B.2 Die Programmierung	261
B.3 Die Initialisierung der Karte	263
B.4 Die Graphik-Ausgabe	267
B.5 Hinweise zur Portierung der Algorithmen	269
C. Das PCX-Format	273
C.1 Das PCX-Datei-Format	273
C.2 Speicherung im PCX-Datei-Format	276
C.3 Einlesen des PCX-Dateiformats	279
D. Musterlösungen	283
E. Beispiel für ein Mustererkennungssystem	303
E.1 Vorwissen – Merkmale	303
E.2 Bildvorverarbeitung	304
E.3 Lernphase	305
E.4 Identifizierung	306
E.5 Schlußbemerkung	308
F. Programme zum Buch	309
F.1 Einführung	309
F.2 Der Disketten-/CD-Inhalt	310
F.3 Compiler-/Linker-Optionen	312
F.4 Die Graphik-Modi 12h und 2Eh	313
F.5 Digitale Halbtonverfahren	317
F.6 Digitale Bildverarbeitung (DBV).....	319
Literaturverzeichnis	323
Stichwortverzeichnis	331

Vorwort

Die Verfahren der Digitalen Bildverarbeitung erschließen aufgrund der immer leistungsfähiger werdenden Rechner immer neue Einsatzgebiete. Im Alltag sind Filterungen in modernen Fernsehgeräten oder automatische Histogrammanipulationen in Video-Kameras nur einige der schon selbstverständlichen Beispiele.

Dieses Buch gibt einen umfassenden Einblick in die Möglichkeiten der Digitalen Bildverarbeitung und beschreibt die Grundlagen für dieses interessante Gebiet. Als optimales Bildverarbeitungssystem wird das menschliche Sehsystem eingangs erörtert. Es wird als Referenz verwendet und dadurch die Defizite bei der Bildaufnahme und Bildwiedergabe aufgezeigt. Dabei wird zwangsläufig auch das Gebiet der graphischen Datenverarbeitung gestreift. Für die Vertiefung dieses an die Bildverarbeitung angrenzenden Bereiches werden zum Beispiel die Bücher "Computer Graphics" [ES87] oder [FvD*90] empfohlen.

Nach dieser gerätetechnischen Einführung werden die einzelnen Routinen der Bildverarbeitung in der Reihenfolge angesprochen, wie sie im menschlichen Sehsystem vorkommen und im Zusammenhang mit einem Mustererkennungssystem eingesetzt werden können.

Filterungen im Ortsbereich und auch im Frequenzbereich stehen neben den Histogrammanipulationen am Anfang dieser Reihe. Speziell das umfangreiche Thema "Fourier-Transformation" kann in diesem Zusammenhang nur auf die in bezug der Bildverarbeitung interessanten Teile angesprochen werden. Eine gut verständliche theoretische Abhandlung der Fourier-Transformation ist unter anderem in [Bri89] zu finden und wird dem interessierten Leser zur Vertiefung empfohlen.

Die Kantendetektion als wichtige Methode zur Bildsegmentierung wird als zentrales Thema der Bildverarbeitung umfassend behandelt. Die weiteren Möglichkeiten der Bildsegmentierung werden in einem getrennten Kapitel noch einmal angesprochen.

Da Bilder immer eine große Datenmenge darstellen und somit viel Platz bei der Speicherung belegen, darf in diesem Zusammenhang ein Kapitel über Datenkompression nicht fehlen. Aktuelle Verfahren wie JPEG oder MPEG werden dabei neben den klassischen Methoden erläutert.

Die Verwendung der Bildinformationen und ein Ausblick in das Gebiet der Mustererkennung bilden den Abschluß des Buchs. Der Einsatz aller dieser Verfahren wird im Anhang in einem konkreten Beispiel für ein Mustererkennungssystem noch einmal angesprochen.

Durch eine Liste der Fachbegriffe aus dem Gebiet der Digitalen Bildverarbeitung mit der jeweiligen Definition sowie durch das umfangreiche Stichwortverzeichnis eignet sich dieses Buch auch als Nachschlagewerk.

Die Grundlage zu diesem Buch bilden die mehrjährigen Erfahrungen aus einem zweisemestrigen Vorlesungszyklus Bildverarbeitung – Mustererkennung an der Fakultät für Informatik der Universität Tübingen.

Der Dank gilt daher allen Mitarbeitern und Studenten des Wilhelm-Schickard-Instituts, Graphisch-Interaktive-System (WSI/GRIS) der Universität Tübingen sowie allen Freunden, die mit kreativer Kritik, Hinweisen und Korrekturlesen an der Gestaltung dieses Buches mitgewirkt haben.

Mit der Portierung der Verfahren auf einen Standard-PC soll jedem Interessenten die Möglichkeit gegeben werden, die einzelnen Routinen selbst zu testen. Die abgedruckten Programme sowie die verfügbare Diskette mit dem kompletten Programmcode bilden die Grundlage dazu. Praktische Beispiele aus den unterschiedlichsten Fachgebieten illustrieren die Wirkungsweise der einzelnen Bildverarbeitungsroutinen und sollen Anregungen für den Entwurf eines eigenen Bildverarbeitungs-/Mustererkennungssystems geben.

Mit diesen Beispielen soll der Leser gleichzeitig ein Gespür für die Auswirkungen der einzelnen Verfahren bekommen und somit die Notwendigkeit des Einsatzes der unterschiedlichen Bildverarbeitungsroutinen erkennen.

Aufgrund des systematischen Aufbaus eignet sich dieses Buch auch sehr gut zum Selbststudium. Die Aufgaben am Schluß der einzelnen Kapitel sollen die wichtigsten Punkte noch einmal vertiefen und zur Selbstkontrolle dienen.

Damit dieses Buch und das zugehörige Programmpaket auch in Zukunft praxisnah und interessant bleibt, bin ich für Anregungen, Hinweise und Wünsche jederzeit dankbar.

Romrod, im Frühjahr 1993, R. Steinbrecher

Vorwort zur elektronischen Version

Mit der Veröffentlichung des Buches "Bildverarbeitung in der Praxis" als PDF-Datei in elektronischer Form ist ein lang gehegter Wunsch in Erfüllung gegangen, der erst mit der Freigabe der Rechte durch den Verlag realisiert werden konnte.

Diese Version gibt im wesentlichen das ursprüngliche Buch in Originalform (Inhalt, Bilder, Seitenumbruch etc.) wieder. Einige Tippfehler wurden korrigiert, Inhalte aktualisiert und ergänzt sowie die Grauwertbilder in besserer Qualität wiedergegeben.

Die Anregungen und Wünsche zu diesem Buch in vielen dutzend Briefen und E-Mails konnten aus Zeitgründen leider noch nicht alle berücksichtigt werden. Mit dieser Version von Ende 2005 ist aber ein Schritt in diese Richtung getan.

Für die Bildverarbeitungsseminare wurde eine komplett neue Testumgebung unter Windows mit zusätzlichen Bildverarbeitungsroutinen entwickelt. Die konstruktiven Vorschläge aus diesen Seminaren wurden bei der Überarbeitung dieser Zeilen und der Programme ebenfalls berücksichtigt. – Für Anregungen bin ich weiterhin sehr dankbar.

Romrod, im Frühjahr 2002 und im Winter 2005, R. Steinbrecher

1. Einführung

Fast in allen Bereichen des täglichen Lebens werden wir direkt oder indirekt mit Verfahren oder Ergebnissen der Digitalen Bildverarbeitung konfrontiert. Dies reicht von dem schon selbstverständlichen Fernsehgerät mit immer mehr eingebauten Bildverarbeitungsfunktionen (digitale Filter, Farb-/Grauwertkorrekturen, Bild im Bild, Zoom usw.), über die aufbereitete Wetterkarte von Satelliten oder der optischen Überwachung von Szenen und der Analyse von Aufnahmen bis hin zu "sehenden" Robotern.

Oft ist erst auf den zweiten Blick ersichtlich, wieviele Verfahren aus der Bildverarbeitung bei der Lösung einer bestimmten Aufgabe eine Rolle spielen. Die Komponenten einer Bildverarbeitungsanlage, wie in Abbildung 1.1 dargestellt, lassen sich aber in diesen Systemen immer wiederfinden. Über ein *Aufnahmegerät* (Kamera, Scanner, Frame-Grabber usw.) wird das Ausgangsbild/-signal in digitale Form überführt. Diese Daten werden in der Regel in einer Matrix gespeichert. Jedes Matrixelement repräsentiert den Grau- oder Farbwert eines bestimmten Bildpunktes. Eine *Verarbeitung* dieser digitalen Signale erfolgt danach in Rechnern, deren Leistungsfähigkeit und Aufbau sich nach der jeweiligen Aufgabe richtet. Auf dem *Ausgabegerät* (Bildschirm, Drucker u.ä.) wird dem Benutzer dann das Resultat präsentiert.

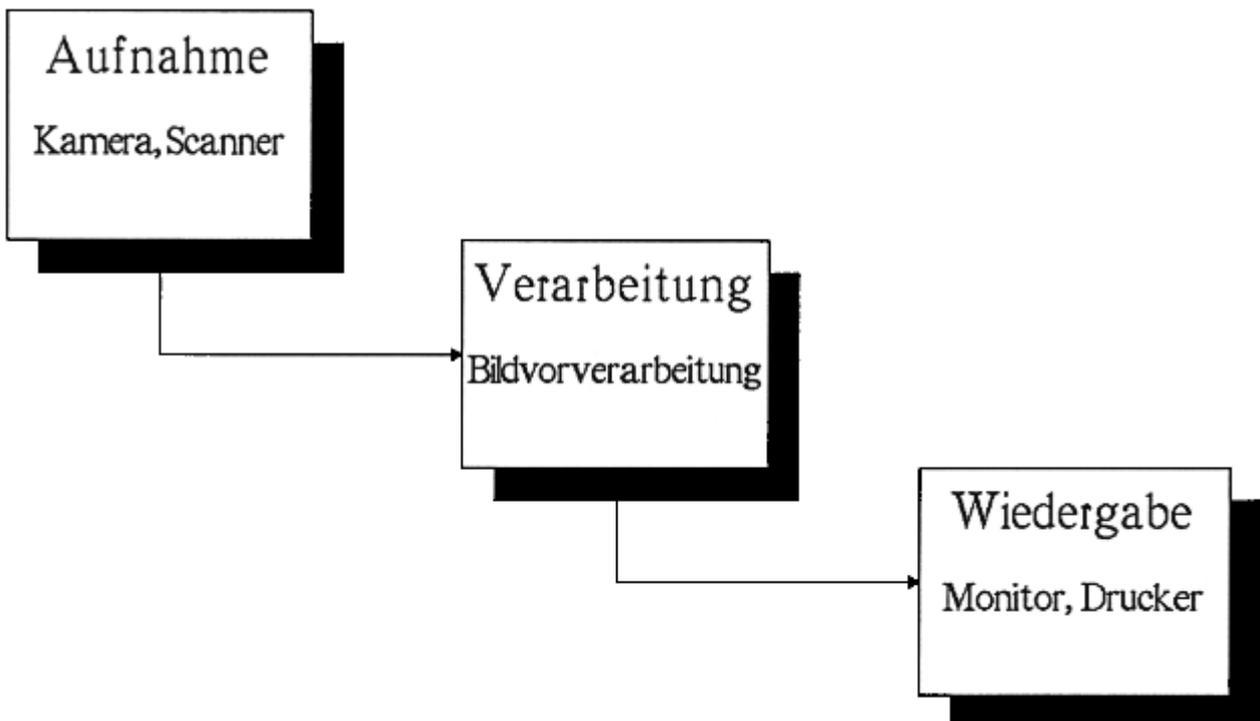


Abbildung 1.1: Der Aufbau eines Bildverarbeitungssystems bestehend aus dem Aufnahmeteil, dem Verarbeitungsteil und dem Wiedergabeteil.

Die Bildverarbeitungsroutinen in einem solchen einfachen System dienen in der Regel dazu, Fehler bei der Aufnahme oder Wiedergabe zu beseitigen oder derart zu vermindern, daß sie vom

1. Einführung

Betrachter nicht mehr wahrgenommen werden. Fehler dieser Art können Rauschen, Farb-/Grauwertfehler, geometrische Verzerrungen usw. sein. In einem weitergehenden Schritt werden bestimmte Eigenschaften oder Strukturen in den Bildern hervorgehoben, um sie dem Betrachter deutlicher sichtbar zu machen. Beispiele dafür sind Kontraständerung, Kantenverstärkung, Kantendetektion und Segmentierung.

Die so kondensierten Merkmale einer Aufnahme werden nicht nur vom Anwender ausgewertet, sondern auch von *Mustererkennungssystemen* interpretiert. Solche Systeme greifen in der Regel auf große Datenbanken zu, mit Informationen über die in Frage kommenden Objekte wie z.B. heuristischen Erfahrungswerten und Regeln, wie und mit welcher Gewichtung die einzelnen gewonnenen Daten (Merkmale) in die Gesamtinterpretation eingehen sollen.

Der Übergang zwischen der reinen Bildverarbeitung und der Bildinterpretation/Mustererkennung ist fließend. Immer muß das Gesamtsystem mit allen Komponenten betrachtet werden, da alle Teile einen Beitrag zum gewünschten Ergebnis leisten. Durch eine direkte Verwertung (Rückkopplung) der Ergebnisse im Aufnahme teil oder für die Verfahren der Bildverarbeitung kann ein System direkt reagieren, um z.B. bei unsicheren Ergebnissen weitere Aufnahmen aus anderen Ansichten anzufertigen oder Bildverarbeitungsalgorithmen mit neuen Parametern anzuwenden.

Die Konstruktion eines "sehenden" Roboters, der sich selbständig durch Fabrikhallen bewegt, Hindernisse umgeht und die gewünschten Maschinenteile aufgrund ihrer Form, Farbe oder bestimmter charakteristischer Merkmale findet, stellt mit den Möglichkeiten der Bildverarbeitung theoretisch kein Problem mehr da.

Mit dieser kurzen Einführung ist zugleich der rote Faden dieses Buches festgelegt. Nach einem historischen Überblick (Kapitel 2) wird zuerst das menschliche Bildverarbeitungssystem betrachtet (Kapitel 3), an dem sich die meisten technischen Realisierungen orientieren.

Danach werden die wichtigsten Geräte zur Bildaufnahme und Bildwiedergabe aufgelistet (Kapitel 4 und 5) und deren Möglichkeiten, Grenzen und Defizite dargelegt. Damit wird gleichzeitig deutlich, wann und weshalb bestimmte Bildverarbeitungsverfahren anzuwenden sind, um Defizite zu vermindern und welche Ergebnisse danach bestenfalls zu erhalten sind.

Die wichtigsten Ergebnisse, also die mit den vorgestellten Verfahren gewonnenen Bilder, sind in diesem Buch als Grauwertbilder abgedruckt. Diese sind mit sogenannten digitalen Halbtonverfahren (Simulation von Graustufen durch unterschiedliche Punktgrößen oder Punktdichten auf binären Ausgabegeräten) erzeugt worden. Zur besseren Bewertung dieser Bilder werden die wichtigsten Halbtonverfahren in Kapitel 6 erläutert.

Sind die Störquellen genau bekannt und lassen sich diese mathematisch beschreiben, kann eine entsprechende Beseitigung der Störungen mit den Verfahren der Bildrestaurierung aus Kapitel 7 erfolgen.

Verfahren, die ein Bildsignal in Signale unterschiedlicher Frequenz und Amplitude zerlegen, bilden einen zentralen Teil der Bildverarbeitung. Mit Hilfe dieser anderen Darstellungsart für ein Bild lassen sich viele Berechnungen und Filterungen einfacher und schneller durchführen. Die Fourier-

Transformation aus Kapitel 8 ist der wohl bekannteste Vertreter dieser Verfahren. Weitere Verfahren werden in Kapitel 9 beschrieben.

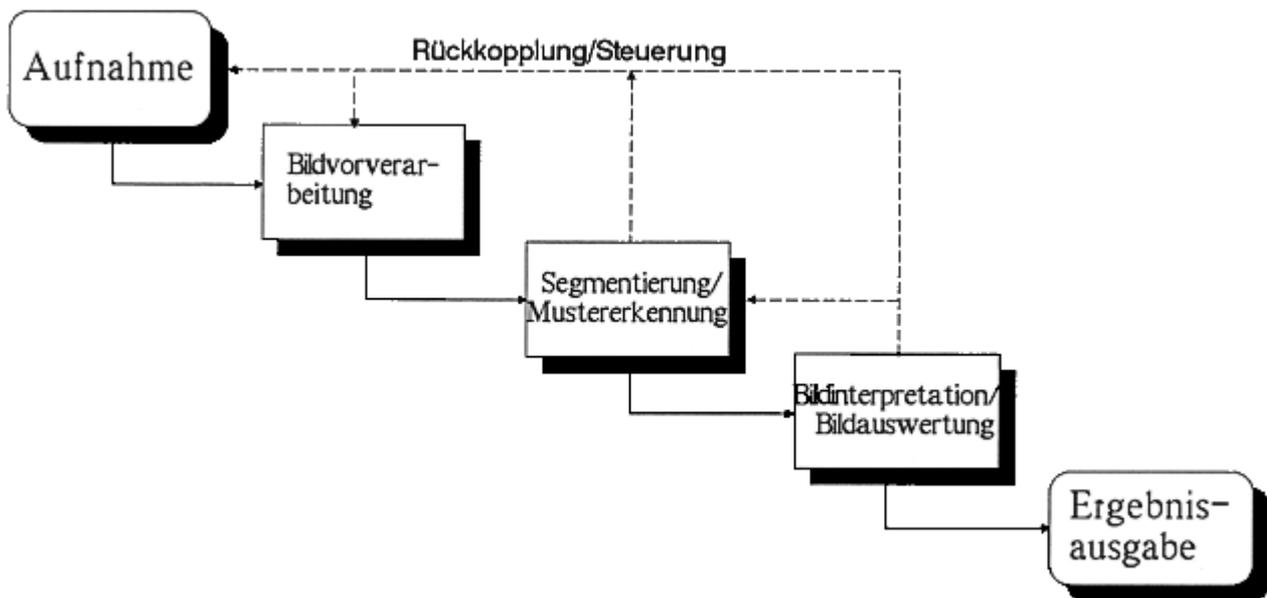


Abbildung 1.2: Der Übergang zwischen der reinen Bildverarbeitung und Mustererkennung bzw. Bildinterpretation ist fließend. Besteht eine Rückkopplung von dem Erkennungs-/Interpretationsteil auf die Aufnahme oder auf die Bildverarbeitung, so kann ein System auf das Ergebnis "intelligent" reagieren.

Im Gegensatz zur Bildrestaurierung dienen die unterschiedlichen Methoden aus Kapitel 10 nur der subjektiven Bildverbesserung. Oft stellen diese aber schon einen wichtigen Schritt zur Mustererkennung dar.

Durch die Kantendetektion (Kapitel 11) mit der in der Regel sich anschließenden Skelettierung (Kapitel 12) wird die große Informationsmenge eines Bildes auf die je nach Anwendung wichtigsten Bestandteile reduziert. Eine Bildinterpretation und Mustererkennung wird damit vereinfacht.

Eine Bildinterpretation wird ebenfalls erleichtert, wenn die Daten nicht nur in Form von einzelnen Bildpunkten, sondern in geometrisch höheren Strukturen vorliegen. Die Vektorisierung (Kapitel 13) ist dabei ein grundlegender Schritt.

Eine Bildsegmentierung kann natürlich auch auf anderem Wege als über die Kantendetektion erfolgen. Die gängigsten Verfahren werden daher in Kapitel 14 erläutert.

Bilder stellen eine enorme Datenmenge dar. Eine einfache Speicherung ist aus Speicherplatzgründen in der Regel nicht möglich. Verfahren der Bildkodierung (Kapitel 15) führen eine Komprimierung durch, die verlustfrei ist oder deren Verluste nicht vom Anwender wahrgenommen werden. Gerade bei der Übertragung von Bilddaten über langsame Medien (Bildtelefon) oder bei Multimedia-Anwendungen ist die Verwendung solcher Verfahren unumgänglich.

1. Einführung

Als Abschluß und als eine Art Ausblick wird im Kapitel 16 das Thema Mustererkennung noch einmal detaillierter angesprochen.

Im Anhang sind neben Begriffsdefinitionen die Beschreibung eines einfachen Bildverarbeitungssystems, die Musterlösungen zu den Aufgaben sowie einige Anmerkungen zur Portierung der vorgestellten Algorithmen zu finden.

2. Historie

Solange der Mensch Bilder erstellt, erfolgt auch nachträglich eine Bearbeitung dieser Werke. Dies kann ein Nachzeichnen von Strukturen in Gemälden, ein Retuschieren von Photographien usw. sein. Sinn und Zweck dieser *Bildbearbeitung* ist in der Regel eine Betonung einzelner Bildteile oder die Beseitigung von Aufnahme Fehlern.

Erst die Umwandlung von Bildinformation in elektrische Signale ermöglichte die (digitale) *Bildverarbeitung*, d.h. die Verarbeitung der gesamten Information aufgrund von mathematischen Vorschriften in Computern. Dabei liegt das Ergebnisbild oft in einer anderen Darstellungsform als die Ausgangsinformation vor, die es erlaubt, bestimmte Bildmerkmale besser sichtbar zu machen.

Mit der Entwicklung des Telegraphen Anfang des 19. Jahrhunderts und der Entdeckung der Photographie durch N. Niepce und L. J. M. Daguerre 1839 kam neben der Textübertragung immer mehr der Wunsch nach der schnellen Bildübermittlung auf. Schon 1866 wurde nach Ideen des italienischen Physikprofessors Giovanni Caselli die erste Maschine gebaut, die Texte und Bilder durch eine zeilenweise Abtastung in elektrische Signale umwandelte und diese über Draht zum Empfänger sendete. Der Absender mußte dafür seine Mitteilung in nichtleitender Tinte auf eine dünne Zinnplatte schreiben bzw. malen. Diese wurde danach auf eine rotierende Trommel gespannt und mittels einer dünnen Nadel mit einer Auflösung von ca. 3 Zeilen pro Millimeter abgetastet. Beim Abtasten des Zinn-Untergrundes wurde ein elektrischer Impuls erzeugt, an den mit Tinte beschriebenen Stellen gab es keinen Impuls. Diese Impulse wurden über Draht dem Empfänger zugeleitet. Die baugleiche Maschine wandelte dort die Impulse wieder in Bilder zurück, indem sie nur bei fehlenden Impulsen auf Papier schrieb. Dieser "Pantelegraph" übermittelte schon im ersten Betriebsjahr über 5000 Unterlagen zwischen Paris und Lyon. Dieses frühe binäre Fax-Gerät enthielt schon alle Merkmale moderner Bildaufnahme geräte.

Der nach R. Hell benannte "Hellschreiber" stellt eine Weiterentwicklung dar. Dieses Gerät überträgt die Information nicht mehr über Draht, sondern über Funk. Um dabei eine möglichst hohe Fehlersicherheit zu erreichen, wird jedes zu übertragende Zeichen in 7×7 Rasterpunkte zerlegt, diese als Stromimpulse nacheinander übertragen und beim Empfänger durch punkweisen Druck auf Papier wieder in lesbare Zeichen zurückverwandelt.

Eine weitere technische Errungenschaft begünstigte die Entwicklung der Digitalen Bildverarbeitung, nämlich die Verbindung von Europa und Amerika durch das transatlantische Unterwasserkabel Anfang der 20er Jahre dieses Jahrhunderts. Neben dem regen Nachrichtenaustausch in Form von Texten kam dabei auch der Wunsch nach der Übertragung von Bildinformation (nicht nur binärer Art) auf. Dafür wurden die Bilder in Zahlenmatrizen transformiert, wobei die einzelnen Zahlen die Grauwerte einzelner Bildpunkte repräsentierten. Die Übertragung dieser Zahlen und die Rückwandlung in Bilder stellte keine Schwierigkeit dar.

Nach diesen ersten wichtigen Schritten gelang der Digitalen Bildverarbeitung der große Durchbruch zur eigenständigen Fachdisziplin durch die Entwicklung von Großrechenanlagen Anfang der

2. Historie

60er Jahre. Beschränkte sich bisher die Bildverarbeitung mangels Speichermöglichkeiten auf die Bearbeitung des gerade abgetasteten Bildpunktes, so konnte nun ein komplettes Bild in digitaler Form gespeichert und bearbeitet werden.

Jedes Bild wird dabei als Matrix bestehend aus $M \times N$ Bildpunkten aufgefaßt. Die Digitale Bildverarbeitung verwendet Verfahren aus anderen Gebieten wie der Optik, der Elektrotechnik, der Nachrichten- und Signaltheorie und der Mustererkennung. Eine rasche Verbreitung dieser Fachdisziplin erfolgte durch die schnelle Entwicklung der Rechnertechnologie und der damit möglichen Problemlösungen in der Medizin, der Astronomie und der Raumfahrt. Mit der Entwicklung von Luftaufnahme-Satelliten erschloß die Raumfahrt digitale Bilddaten einer breiten Anwendung in den zivilen Bereichen von Bodenschatzsuche, Ernteüberwachung, Raumplanung, Umweltschutz und vor allen Dingen der Wettervorhersage.

Für spezielle Anwendungen werden heutzutage immer wieder neue Digitale Signalprozessoren (DSP) zur schnellen Filterung entwickelt. Damit wird eine Bildverarbeitung und Bildanalyse selbst größerer Bilder in Echtzeit, d.h. in der Geschwindigkeit der Bildaufnahme ermöglicht.

3. Das menschliche Bildverarbeitungssystem

Das beste bisher bekannte Bildverarbeitungs- und Mustererkennungssystem ist immer noch das menschliche Auge in Verbindung mit der Bildauswertung durch das Gehirn. Viele technische Systeme orientieren sich am Aufbau und der Funktionsweise des Auges. Als Fernziel von Bildverarbeitungssystemen wird immer wieder die Leistung des biologischen Vorbildes anvisiert.

Zum besseren Verständnis der einzelnen Verfahren in der Bildverarbeitung soll zuerst das Auge, sein Aufbau und die prinzipielle Funktion näher erklärt werden. In vielen Fällen werden die Ergebnisse dieser Verfahren im letzten Schritt durch den Menschen ausgewertet. Dazu müssen die Ausgabedaten der Bildverarbeitungsroutinen in einer dem Menschen verständlichen Form dargestellt werden. Die Randbedingungen dafür sind wieder durch die technischen Daten des menschlichen Bildverarbeitungssystems gegeben.

Unter diesem Aspekt werden bestimmte Vorgehensweisen in der Bildverarbeitung einleuchtend, Vereinfachungen verständlich und die Aussagekraft von Ergebnissen relativiert.

3.1 Der Aufbau des Auges

Bildaufnahme und Bildverarbeitung lassen sich beim menschlichen Sehsystem nicht strikt trennen, da bereits direkt bei der Bildaufnahme verschiedene Verarbeitungsschritte (Filterungen) stattfinden.

Beim Menschen haben die Augen einen Durchmesser von ca. 20 mm und bestehen aus dem Augapfel und seinen Anhangsgebilden (Augenlider, Tränendrüsen, Bindehaut usw.), die der Bewegung, dem Schutz und der Versorgung des Auges dienen.

Der Augapfel liegt, in Fettpolster eingebettet, in der Augenhöhle. Der Sehnerv bildet die Verbindung zum Gehirn. Die Hülle des Augapfels besteht aus drei Schichten. Die äußere Hülle des Auges ist die Lederhaut (*Sclera*), die vorne in die durchsichtige Hornhaut (*Cornea*) übergeht. Darunter liegt die gut durchblutete Aderhaut (*Chorioidea*), an die sich innen die Netzhaut (*Retina*) mit den Stäbchen (Hell-Dunkel-Sehen) und den Zapfen (farbiges Sehen) anschließt. Die Netzhaut nimmt die einfallenden Lichtreize auf. Der Sehnerv leitet die entsprechenden Erregungen zum Gehirn weiter [MBM81].

Die vordere Augenkammer wird hinten durch die ringförmige Regenbogenhaut (*Iris*) begrenzt, die sowohl aus Teilen der Aderhaut als auch der Netzhaut gebildet wird. Die Pupille ist eine größenveränderliche Öffnung (2-8 mm) in der Iris, durch die unter anderem die Lichteintrittsmenge reguliert wird. Hinter der Pupille liegen Linse und Glaskörper. Die Linse besteht aus konzentrischen Lagen von faserartigen Zellen und enthält ca. 60-70% Wasser, 6% Fett, Proteine usw. Die Linse ist durch eine gelbe Pigmentation leicht eingefärbt. Diese Einfärbung nimmt mit steigendem Alter zu.

3. Das menschliche Bildverarbeitungssystem

Infrarotes und ultraviolettes Licht werden durch Proteine in der Linse absorbiert. Eine intensive Strahlung aus diesem Bereich kann das Auge schädigen.

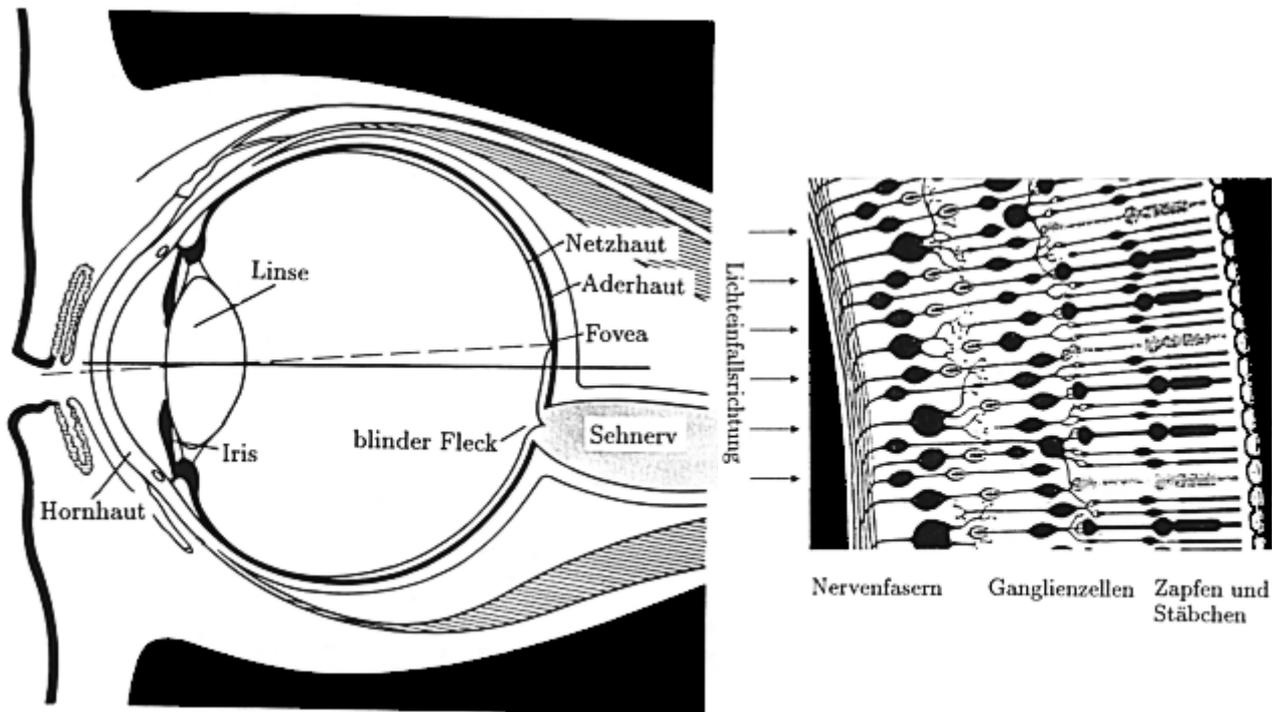


Abbildung 3.1: Der Aufbau des menschlichen Auges. Links: Querschnitt durch den Augapfel. Rechts: Querschnitt durch die Netzhaut.

Auf der Netzhaut entsteht ein umgekehrtes, reelles Bild, das durch den Sehnerv und die Sehbahnen zum Sehzentrum des Großhirns geleitet wird. Räumliches Sehen wird durch Vermischen, der von beiden Augen gelieferten Bilder ermöglicht.

Die Eintrittsstelle des Sehnervs in den Augapfel ist nicht lichtempfindlich (*blinder Fleck*). Schläfenwärts davon liegt der gelbe Fleck (*Fovea*), die Stelle des schärfsten Sehens, da hier die Dichte von Zapfen am größten ist. Das Nah- und Fernsehen geschieht durch Änderung der Linsenform durch den Ziliarmuskel, der sich rund um die Linse befindet. Kontrahiert sich der Ziliarmuskel, nimmt die Linse Kugelform an, die Brechkraft erhöht sich und nahe Gegenstände werden scharf abgebildet.

3.2 Die lichtempfindlichen Teile des Auges

Die lichtempfindlichen Sinneszellen (Zapfen und Stäbchen) befinden sich in der äußersten, an das Pigmentepithel angrenzenden Schicht der Netzhaut. Das Licht muß also, außer in der Fovea, alle übrigen Schichten durchdringen, ehe es auf die eigentlichen Rezeptoren trifft.

Pro Auge besitzt der Mensch ca. 6-7 Millionen farbempfindliche Sensoren, die Zapfen. Diese sind in der Hauptsache in und um die Fovea verteilt. Die Dichte beträgt hier ca. 160000 Zapfen pro

3.2 Die lichtempfindlichen Teile des Auges

mm² (beim Bussard sogar über 1000000) und fällt zum Netzhautrand hin bis auf 5000 ab. Die Zapfen kann man noch einmal in drei Gruppen mit unterschiedlicher spektraler Empfindlichkeit einteilen: rote, grüne und blaue Zapfen. Jede dieser Rezeptorengruppe und auch die Stäbchen werden in ihrer Empfindlichkeit getrennt geregelt. Dabei kann die Intensität einer Gruppe bis auf ein 1 / 1000 reduziert werden. Die maximale Empfindlichkeit liegt bei einer Wellenlänge von etwas 555 nm. Da jeder dieser Zapfen in der Fovea mit bis zu drei Nervenfasern verbunden ist, kann der Mensch sehr feine Details mit diesen Sinneszellen erfassen. Auge und Kopf werden dabei automatisch so ausgerichtet, daß die Objekte des Interesses auf diesen gelben Fleck abgebildet werden.

Die kleinsten Zapfen haben einen Durchmesser von 1 µm. Zum Rand hin sind die Rezeptoren zunehmend zusammenschaltet, um die Empfindlichkeit auf Kosten der Auflösung zu erhöhen.

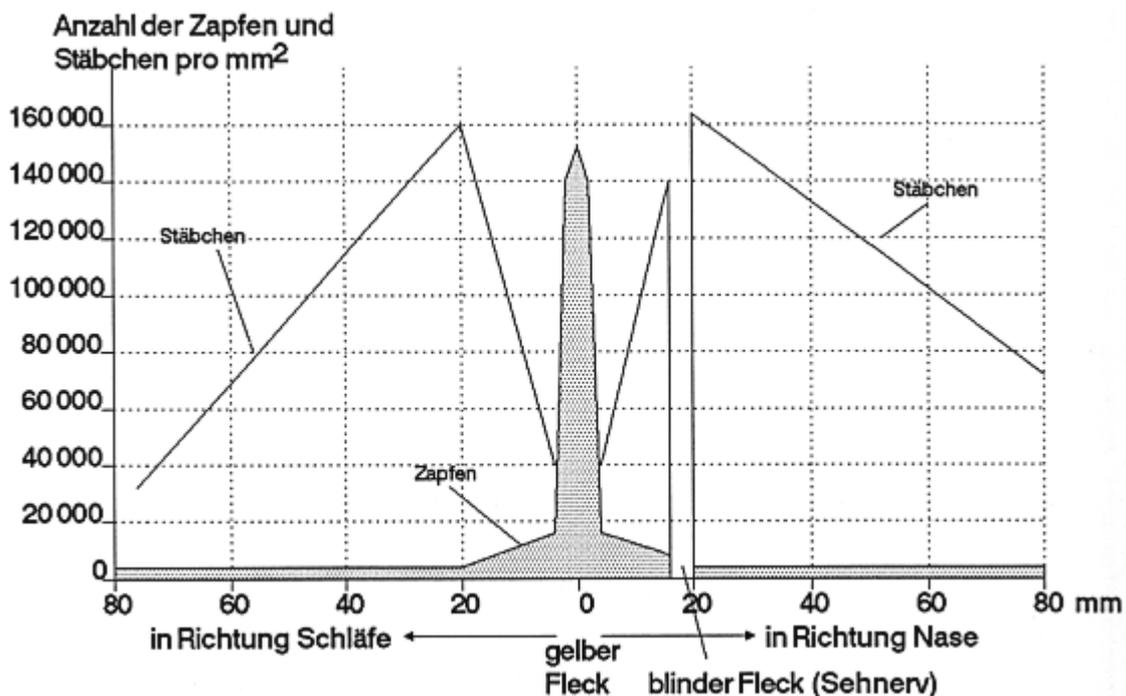


Abbildung 3.2: Dichte der Zapfen und Stäbchen auf der Netzhaut. Die Dichte der Zapfen ist am gelben Fleck am größten und nimmt zum Rand hin stark ab.

Die Stäbchen dagegen sind, bis auf die Fovea-Fläche, über die ganze Retina verteilt. Ihre Anzahl beträgt ca. 75-150 Millionen. Dabei sind immer mehrere dieser hell-/dunkel-empfindlichen Stäbchen (im Durchschnitt 130) mit einem Nerv verbunden.

3.2.1 Signalauswertung und Weiterleitung

Die lichtempfindlichen Zellen in der Netzhaut setzen die optischen Reize in Nervensignale um und leiten die Impulse zu den Ganglienzellen (= Ansammlung von Nervenzellen) der Netzhaut weiter.

3. Das menschliche Bildverarbeitungssystem

Stäbchen und Zapfen reagieren auf Licht mit Spannungs- und Stromänderungen. Die Spannung kann dabei um bis zu 25 mV pro Sinneszelle, der Strom um bis zu 30 pA schwanken. Chemisch sind daran Membrane beteiligt, die ihre Durchlässigkeit für Natrium-/Kaliumionen in Abhängigkeit des Lichtes ändern. Die Stäbchen haben beim Menschen eine Ansprechzeit von ca. 300 msec. Zapfen hingegen reagieren auf einen Lichtimpuls schon nach 80 – 90 msec.

Experimente haben gezeigt, daß das Auge nur Informationen über Lichtänderungen an Grenzlinien an das Gehirn weiterleitet. Bereiche für die keine Änderungen gemeldet werden, ergänzt das Gehirn als gleichförmig (vergleiche Kapitel 15). Um solche Änderungen an Grenzlinien entdecken zu können, führen die Augen ständig kleine Zitterbewegungen (Sakkadische Augenbewegung; *saccade frz.* = Ruck) aus und lassen somit das Bild des Gegenstandes auf der Netzhaut hin und her wandern. Bis zu diesem Punkt sind die Signale also bereits bis zu einem gewissen Grad analysiert und vorverarbeitet worden.

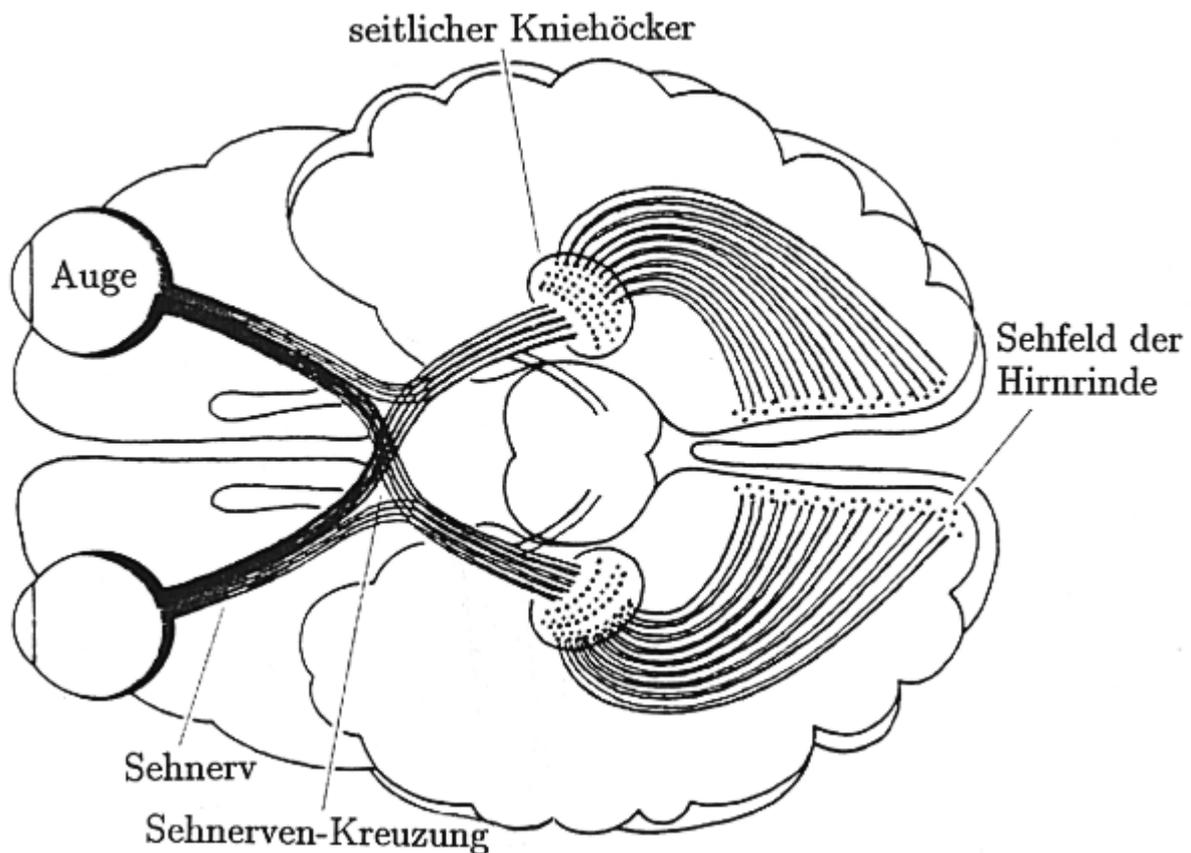


Abbildung 3.3: Der Verlauf der Sehnerven von den Augen zum Sehfeld der Hirnrinde. In den einzelnen Verarbeitungsstufen von der Netzhaut bis zum Sehfeld erfolgt eine unterschiedliche Verarbeitung/Abstraktion und Reaktion auf die Signale. So reagieren bestimmte Nervenzelle im Sehfeld der Hirnrinde zum Beispiel vorzugsweise auf die Bewegung des Bildes auf der Netzhaut in einer bestimmten Richtung.

Die Nervenfasern der Ganglienzellen (etwa eine Million pro Netzhaut) bilden zusammen den Sehnerv. Ein großer Teil der Fasern zieht ohne Unterbrechung zu zwei Zellanhäufungen tief ins Gehirn, den äußeren Kniehöckern, in denen wahrscheinlich die räumlichen Unterschiede empfindlicher als auf der Netzhaut registriert werden, und bilden dort Synapsen, also Kontaktstellen zwi-

schen ableitenden und aufnehmenden Nerven. Die Zellen der Kniehöcker sind durch ihre Nervenfasern unmittelbar mit dem primären Sehfeld verbunden. Hier findet eine Punkt-für-Punkt-Abbildung der von der Netzhaut weitergeleiteten Signale statt. Vom primären Sehfeld aus werden die Informationen über mehrere Synapsen zu benachbarten Rindengebieten und an tief im Gehirn gelegene Stellen weitergeleitet. Einige Nervenfasern ziehen sogar vom primären Sehfeld zurück zu den Kniehöckern. Welche Funktion diese Rückkopplung hat, ist noch unbekannt. Das primäre Sehfeld ist also keineswegs das Ende sondern nur eine Stufe in der Verarbeitung der visuellen Information.

Besonders auffällig ist die teilweise Überkreuzung der Sehnerven. Dadurch wird der rechte Teil des Gesichtsfeldes, also dasjenige Gebiet, welches das ruhende Auge von der Außenwelt erfaßt, auf das linke Sehfeld abgebildet. Kniehöcker und Sehfeld der linken Hirnhälfte empfangen Informationen von den beiden linken Netzhauthälften, die ihrerseits die rechte Hälfte des Gesichtsfeldes erfassen. Entsprechend übernehmen der rechte Kniehöcker und das rechte Sehfeld die linke Hälfte des Gesichtsfeldes. Erst durch diese Überkreuzungen und das Vermischen der Information beider Augen ist räumliches Sehen möglich.

3.2.2 Technische Daten des menschlichen Sehsystems

Das menschliche Auge enthält rund 6-7 Millionen Zapfen zum Farb-Sehen und etwa 75-150 Millionen Stäbchen zum Schwarzweiß-Sehen. Zapfen reagieren auf Informationsänderungen in etwa 80-90 ms, Stäbchen in rund 300 ms.

Je nach Lage im Auge, werden die Signale einer verschiedenen Anzahl von Zapfen bzw. Stäbchen zusammengefaßt und schon in einer gewissen Weise vorverarbeitet. Die Reize werden dann über den Sehnerv, der selbst rund 1 Million Nervenfasern besitzt, zum Gehirn weitergeleitet.

Durch Veränderung der Pupille kann die auf das Auge fallende Lichtmenge um das 16fache vermindert werden. Durch Änderung der Empfindlichkeit der Rezeptoren, kann sich das Auge an einen Helligkeitsbereich von $1 : 10^{10}$ anpassen. Die subjektive Helligkeit ist dabei eine logarithmische Funktion der Lichtintensität.

Der maximale räumliche Sehwinkel bei ruhendem Auge beträgt ca. +/-30 Grad zur Augenachse, wobei nur in der Nähe von 0 Grad scharf und deutlich gesehen wird. Dieses wahrnehmbare Gebiet wird auch als *Gesichtsfeld* bezeichnet. Am Rand erkennt man Objekte und Bewegungen nur schemenhaft. Durch Bewegen der Augen bei ruhiggestelltem Kopf kann man einen Bereich von mehr als 180 Grad überstreichen (*Blickfeld*).

Das Auflösungsvermögen ist zum einen durch die Größe der Zapfen und Stäbchen, zum anderen durch den Kontrast bestimmt. Bei optimalem Kontrast kann man noch eine Raumfrequenz von 50 Perioden pro Sehwinkelgrad unterscheiden. Das Optimum für die Kontrastempfindlichkeit liegt bei etwa 3 Perioden pro Sehwinkelgrad. Dies entspricht etwa der Auflösung eines 12 DIN-Films.

Da die Zapfen und Stäbchen nicht mit binären Werten sondern mit kontinuierlichen, analogen Werten auf Reize reagieren, können auch Objekte registriert werden, die in der Größe unter der

3. Das menschliche Bildverarbeitungssystem

theoretischen Auflösung der lichtempfindlichen Zellen liegen. Dieser Effekt wird als *Überauflösung* bezeichnet.

Horizontale und vertikale Strukturen werden durch das Auge bevorzugt. Diagonale Verläufe registriert das Auge mit einer um ca. 10% verminderten Intensität [CM87].

Das gesunde menschliche Auge ist in der Lage, etwa 160 Farben voneinander zu unterscheiden. Werden Grautöne, Sättigung und Helligkeiten mitberücksichtigt, so lassen sich etwa 7 Millionen Farbtöne differenzieren [Bö87]. Bei reinen Schwarzweißbildern vermag das Auge auch nur ca. 30 Grautöne zu unterscheiden [Nie82].

Die Signalauswertung erfolgt nicht an einer Stelle allein. Die Information wird stufenweise zerlegt (Zapfen, Stäbchen, Sehnerven, Sehnerven-Kreuzung, Kniehöcker, Sehfeld usw.) und dabei abstrahiert. Dieser Aufbau gibt schon erste Hinweise darauf, wie ein technisches Bildverarbeitungssystem aufgebaut sein sollte.

Aufgaben

Aufgabe 1

Die Zapfen der Retina beim menschlichen Auge (Durchmesser ≈ 20 mm) haben einen Durchmesser von ≈ 5 μm . Berechnen Sie den Sehwinkel, der damit theoretisch maximal durch das Auge aufgelöst werden kann.

Wie erklären Sie sich, daß sich bei Versuchen herausgestellt hat, daß der Mensch noch räumliche Verschiebungen von $10''$ (Winkelsekunden) auflösen kann. Begründen Sie den Unterschied dieser beiden Ergebnisse.

Aufgabe 2

Das menschliche Auge kann nur eine bestimmte Anzahl Farben und Grautöne unterscheiden. Wie groß ist die jeweilige Anzahl etwa? Welche Konsequenzen hat dies für die Bildverarbeitung sowie für die Bildwiedergabe?

Aufgabe 3

Die etwa kreisförmige Fovea hat einen Radius von 400 μm . An diesem Fleck des schärfsten Sehens befinden sich rund 160000 Zapfen pro mm^2 . Ein runder Textausschnitt, der aus einem Betrachtungsabstand von 50 cm angeschaut wird, wird komplett auf die Fovea abgebildet.

1. Wie groß ist der Originaltext (Augendurchmesser 20 mm)?
2. Mit welcher Auflösung (Punkte/Inch; dpi) wird der Text vom Auge wahrgenommen?
3. Vergleichen Sie dieses Ergebnis mit der Auflösung älterer Laserdrucker (300 dpi).

4. Geräte zur Bildaufnahme

Die Umwandlung der Helligkeits- und Farbwerte einer Szene in elektrische Impulse geschieht mit Hilfe einer Kamera, eines Scanners o.ä. Das Bild wird bei allen Verfahren in ein feines Raster zerlegt (*Diskretisierung*) und die einzelnen, in Stufen zerlegten (*Quantisierung*) elektrischen Werte dieser Rasterpunkte zur Verarbeitung weitergeleitet (siehe Abschnitt 4.3 und Abschnitt 4.4.3). Je nach Aufbau besitzen die einzelnen Systeme zur Bildaufnahme verschiedene Vor- und Nachteile.

4.1 Die Kameraröhre

Der bekannteste Kamerateyp ist der der Fernsehkamera in der Ausführung der Aufnahmeröhre als *Vidicon*. Über ein photographisches Objektiv (Linsensystem) mit geeigneter Brennweite wird das Bild auf die Frontplatte (*Photokathode*) der Aufnahmeröhre abgebildet. In der photoleitenden Schicht auf der Frontplatte erfolgt die Umsetzung des optischen Bildes in ein elektrisches Ladungsbild. Mittels eines feinen Elektronenstrahls wird das Ladungsbild zeilenweise abgetastet. Die Strahlführung erfolgt dabei durch die Magnetfelder zweier senkrecht zueinander stehender Spulenpaare, die von sägezahnförmigen Ablenkströmen durchflossen werden.

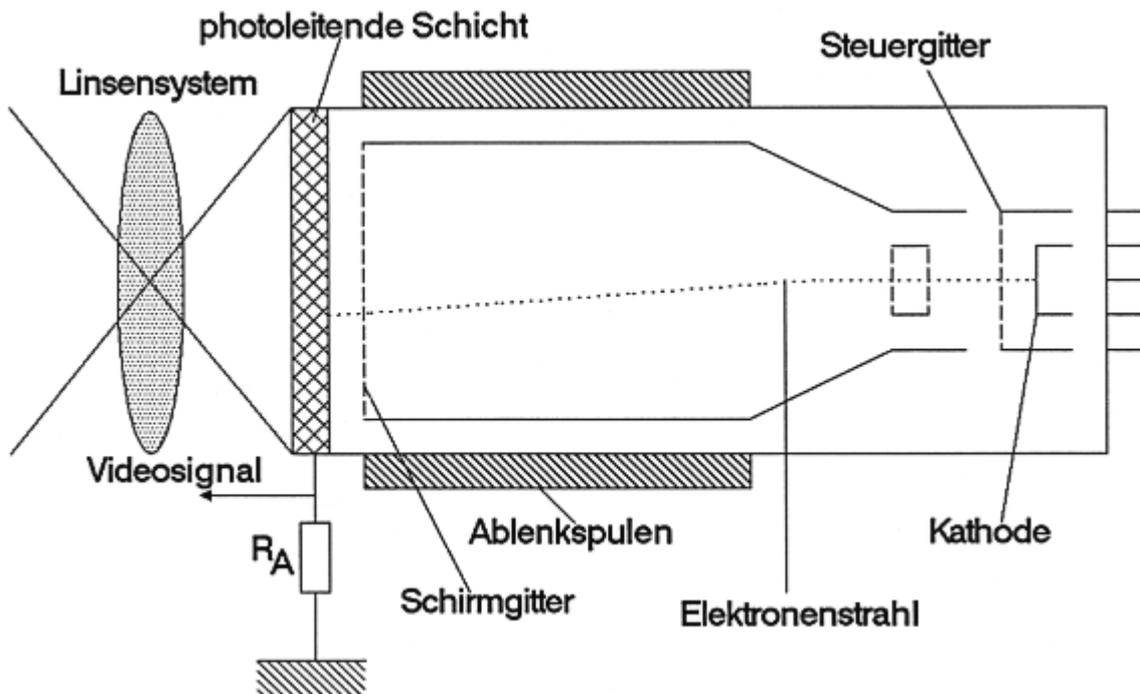


Abbildung 4.1: Schematisierte Querschnittsdarstellung des Aufbaus einer Kameraröhre.

Von außen gesehen besteht eine Kameraröhre hinter dem Linsensystem aus einem optisch planen Glasfenster, auf dessen Innenseite eine transparente, leitende Schicht aufgebracht ist. Auf dieser

4. Geräte zur Bildaufnahme

leitenden Schicht befindet sich als eigentliches Speicherelement eine dünne, photoelektrisch leitende Schicht (siehe Abbildung 4.2). Diese Schicht besitzt bei Dunkelheit einen hohen spezifischen Widerstand, der aber mit steigender Beleuchtungsstärke abnimmt. Die Speicherschicht kann man sich in viele einzelne, voneinander unabhängige Speicherelemente, die jeweils aus einer Parallelschaltung einer Kapazität C_P und eines Photowiderstandes R_P bestehen, vorstellen. Legt man an die Photokathode eine Spannung an und tastet die Speicherschicht bei Dunkelheit mit einem Elektronenstrahl ab (*Dunkelstrom*), so werden die Kondensatoren C_P aufgeladen. Wenn nun ein Bild auf die Speicherplatte abgebildet wird, wird infolge der Beleuchtung das Material der Speicherschicht photoelektrisch leitend, und die einzelnen Speicherelemente werden entsprechend der vorhandenen Beleuchtungsstärke teilweise über die zugehörigen Widerstände R_P entladen. Auf der gesamten Speicherplatte entsteht also eine Ladungsverteilung, die der Helligkeitsverteilung des Bildes entspricht. Durch den abtastenden Elektronenstrahl werden die Speicherelemente wieder auf Kathodenpotential aufgeladen. Der daraus resultierende kapazitive Strom über die Signalelektrode verursacht einen Spannungsabfall am Arbeitswiderstand R_A . Diese Spannung geht als Videosignal zum Vorverstärker.

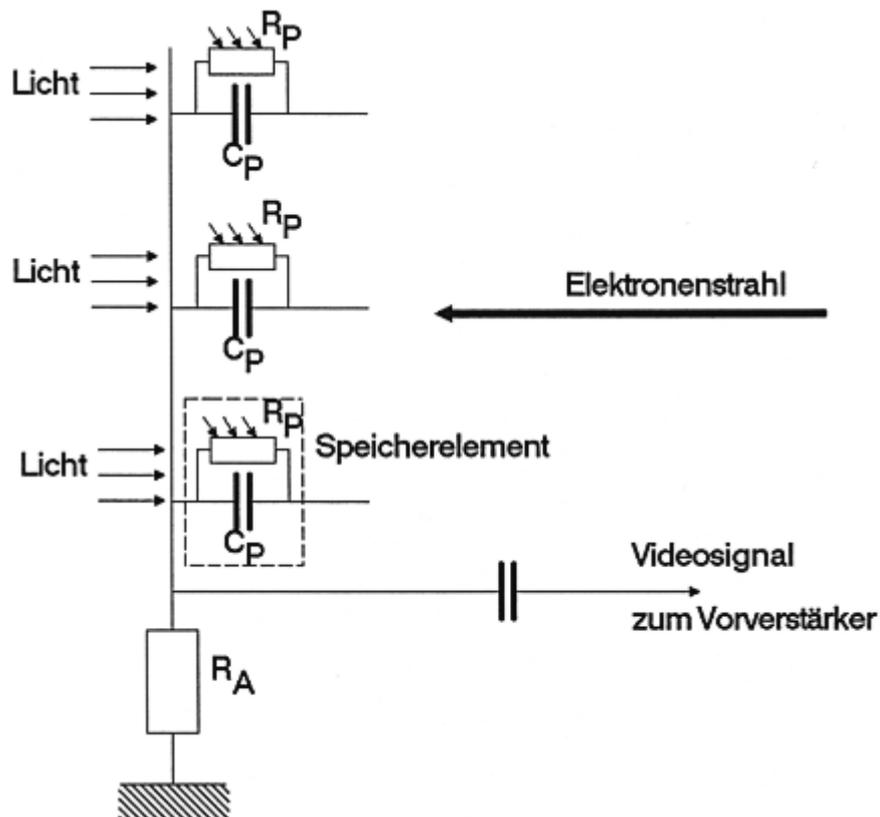


Abbildung 4.2: Prinzipieller Aufbau der lichtempfindlichen Schicht der Kameraröhre. Jeder Bildpunkt kann als eine Parallelschaltung einer kleinen Kapazität und eines Photowiderstandes angesehen werden.

Je nach verwendeter Speicherschicht unterscheidet man verschiedene Arten von Kameraröhren, deren Unterschiede in der Hauptsache in der spektralen Empfindlichkeit liegen. Die jeweils hervorstechendsten Eigenschaften zeigt die folgende Auflistung:

- **Vidicon-Röhren** sind universelle Bildaufnahmeröhren. Die Speicherschicht besteht aus Antimontrisulfid (Sb_2S_3) und gewährleistet einen Dynamikbereich bis 1 : 10000.
- **Silizium-Vidicon-Röhren** weisen eine Spektralempfindlichkeit bis in den nahen Infrarotbereich auf.
- **Plumbicon-Röhren** haben Bleioxyd (PbO) als Speicherschicht und zeichnen sich durch ein besonders geringes Trägheitsverhalten (Nachzieheffekt) aus.
- **Newvicon-Röhren** sind sehr lichtempfindlich. Die spektrale Empfindlichkeit des Photoleiters Zink-Selenid/Zink-Cadmium-Tellurid ($\text{ZnSe} / \text{ZnCdTe}$) reicht bis in den nahen Infrarotbereich.
- **Pasecon-Röhren** mit einer Cadmiumselenid-Beschichtung (CdSe), sind hochempfindliche Kameraröhren mit einer hohen Signalgleichmäßigkeit und Auflösung.
- **Saticon-Röhren** zeichnen sich durch geringe Trägheit, hohe Auflösung und Signalgleichmäßigkeit aus. Die Speicherschicht besteht in der Hauptsache aus Selen (Se-As-Te).
- **Pyroton-Röhren** } sind für Wärmebilder im Infrarotbereich von 2 – 4 μm einsetzbar.
- **SIT- und EIC-Röhren** mit integriertem Bildverstärker sind für den Einsatz bei geringen Beleuchtungsstärken bis 10^{-4} lx geeignet.

Arbeitete man bei den ersten Fernsehversuchen in den 30er Jahren noch mit Auflösungen von 30, 48, 60, 90 und 180 Zeilen, so haben moderne Kameras eine theoretische Auflösung von über 1000 Zeilen und Spalten.

Die zukünftige Fernsehnorm *HDTV* (High-definition Television) sieht eine Auflösung von 1280×720 bis zu 1920×1250 Bildpunkten vor.

Farbkameras gibt es seit Anfang der 50er Jahre. Sie arbeiten genauso wie die hier beschriebene Monochromkamera. Das farbige Bild wird dabei in der Kamera optisch in die drei Farbauszüge der Primärfarben Rot, Grün und Blau zerlegt, drei Aufnahmeröhren zugeführt und dort, wie oben beschrieben, weiterverarbeitet.

4.2 Die CCD-Kamera

Die ersten Versuche mit CCD-Kameras wurden Anfang der 70er Jahre in den Forschungslabors gemacht. Aufgrund der kleineren Abmessungen im Vergleich zu der normalen Kameraröhre sowie der robusteren und leichteren Handhabung hat diese Kameraart eine schnelle Verbreitung gefunden.

Die CCD-Kamera besteht bis auf die Speicherschicht aus denselben Komponenten wie die normale Kameraröhre. CCD steht für *Charge Coupled Device* und erklärt die Art, wie die Lichtintensität in eine elektrische Größe umgewandelt wird. Die Funktionsweise hat große Ähnlichkeit mit

4. Geräte zur Bildaufnahme

der eines dynamischen Speicherchips. Man stelle sich die Speicherzellen homogen auf der Chip-ebene verteilt vor. Die Speicherkondensatoren

C_P werden durch den Photostrom kleiner Siliziumphotodioden D_P geladen, welche an der Chip-oberfläche sitzen und durch ein Fenster geschützt sind. Das "Auslesen" der Zellen (beleuchtetes Feld) geschieht zyklisch, wobei in der Zeit des vertikalen Austastimpulses (bei der Fernsehnorm die Zeit des Strahlrücklaufs von der rechten unteren zur linken oberen Ecke) ein Transfer der bis dahin gesammelten Ladung in ein zweites Feld von Kondensatoren erfolgt (*Frametransfer*). Solange nun die einzelnen Zellen ausgelesen werden, kann die Ladung für das nächste Bild bei anderen Zellen bereits wieder integriert werden.

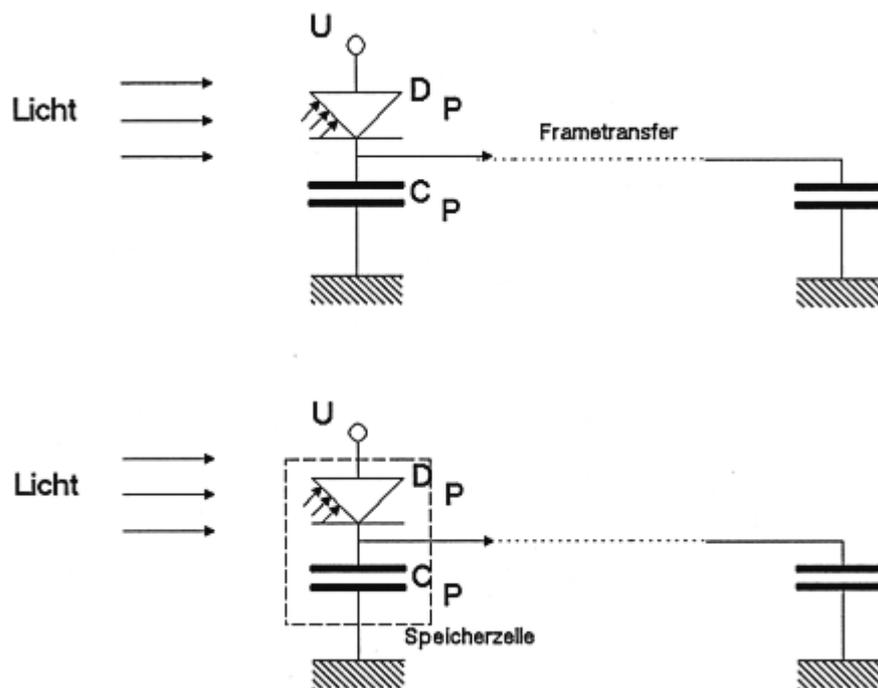


Abbildung 4.3: Die einzelnen Speicherzellen bestehen bei der CCD-Kamera aus einem Kondensator und einer Photodiode, über die der Kondensator bei Lichteinfall aufgeladen wird. Während des vertikalen Strahlrücklaufs wird das komplette Bild ausgelesen und in ein zweites Feld mit Speicherelementen übertragen (Frametransfer).

Man unterscheidet zwei Typen von CCD-Kameras. Die *Zeilenkamera* enthält nur eine Linie mit Speicherzellen auf der Chipoberfläche. Ein Bild entsteht erst durch eine relative Bewegung der Kamera zum Objekt (siehe Abschnitt 4.3). Aufgrund der kleineren Fläche einer solchen Zeilenkamera ist die Wahrscheinlichkeit, daß Verschmutzungen beim Herstellungsprozeß die lichtempfindlichen Zellen beeinträchtigt haben, geringer. Dadurch lassen sich fehlerfreie CCD-Zeilen mit über 8192 Bildpunkten herstellen.

Die *Matrixkamera* besitzt eine rechteckige Matrix von Speicherzellen. Die Pixelanzahl über die gesamte Chipfläche beträgt 30×30 und über 2048×1024 . Die Spitze der Entwicklung stellen derzeit Chips mit 3072×2048 Pixel (= 6.2 Millionen Bildpunkte) dar. Jedes dieser quadratischen lichtempfindlichen Pixel hat dabei eine Kantenlänge von $4.3 \mu\text{m}$, die Chipfläche beträgt über 120mm^2 .

Die Schwierigkeit bei der Herstellung solcher hochauflösenden CCD-Sensoren besteht darin, die Sauberkeit des Herstellungsprozesses zu gewährleisten. Schon kleinste Staubkörner können zum Ausfall von Bildpunkten führen. Im Gegensatz zu Speicherchips kann keine andere Zelle die Aufgabe einer defekten übernehmen.

Mit einem technischen Trick kann jedoch die Auflösung noch erhöht werden. Dabei werden die CCD-Zellen immer um einige Nanometer hin- und herbewegt, um lichtempfindliche Zellen zwischen den eigentlichen CCD-Zellen zu simulieren. Als Stellglieder dienen hier Piezo-Quarze. Durch entsprechende Differenzbildung der einzelnen Signale erhält man so eine theoretische Auflösung von bis zu 21 Millionen (Farb-) Punkten (2994×2320 Punkte pro Farbkanal). Dies entspricht der Qualität eines Farb-Diapositivs. Eine Aufnahme bei dieser hohen Auflösung benötigt allerdings bis zu 8 Sekunden [Len89].

4.3 Der Scanner

Unter einem *Scanner* wird ein Gerät verstanden, das Bilder oder Bildvorlagen zeilenweise bzw. punktweise abtastet. Diese einzelnen Meßwerte werden dann im Rechner durch geeignete Hard- oder Software zu einem kompletten Bild zusammengesetzt. Generell gibt es zwei unterschiedliche Funktionsweisen von Scannern:

- **Scantable, Flachbettscanner**

Das Objekt wird durch einen sich relativ zum Objekt bewegenden lichtempfindlichen Sensor abgetastet.

Diese Art von Scannern wird in der Regel für die Abtastung von 2D-Objekten (Zeichnungen, Photographien, Texte usw.) verwendet. Eine CCD-Zeilenkamera mit hoher Auflösung (bis zu 2048 Punkte pro Inch; per Software bis zu 4096 Punkte) tastet die Vorlage zeilenweise ab, in der Regel mit einer Grauwert-/Farbtiefe von 8-Bit (256 Stufen). Werden drei CCD-Zeilen mit unterschiedlicher spektraler Empfindlichkeit verwendet (für Rot, Grün und Blau), so können Farbbilder mit über 16 Millionen Farbtönen digitalisiert werden.

- **Laserscanner, Flying-Spot**

Das gesamte Objekt wird durch eine Optik auf einen Detektor abgebildet. Die Abtastung erfolgt dann durch punktweise Beleuchtung des Objektes.

3D-Objekte und räumliche Szenen werden mit solchen Scannern abgetastet. Meist wird in diesen Fällen nur ein monochromes Bild gewonnen, dann jedoch zusätzlich mit Tiefeninformation (z-Koordinate).

4. Geräte zur Bildaufnahme

Eine weitere Art der Bilddigitalisierung kann mit sogenannten *Frame-Grabbern* erfolgen. Sie sind als Zusatzkarten für verschiedene Rechner oder auch als eigenständige Geräte verfügbar und tasten das analoge Signal einer Videoquelle (Kamera, Rekorder o.ä.) in konstanten Zeitschritten ab (*Diskretisierung*). Der analoge Wert des Signals zu den entsprechenden Abtastzeiten wird durch einen Analog/Digital-Konverter in ein digitales Signal umgewandelt (*Quantisierung*).

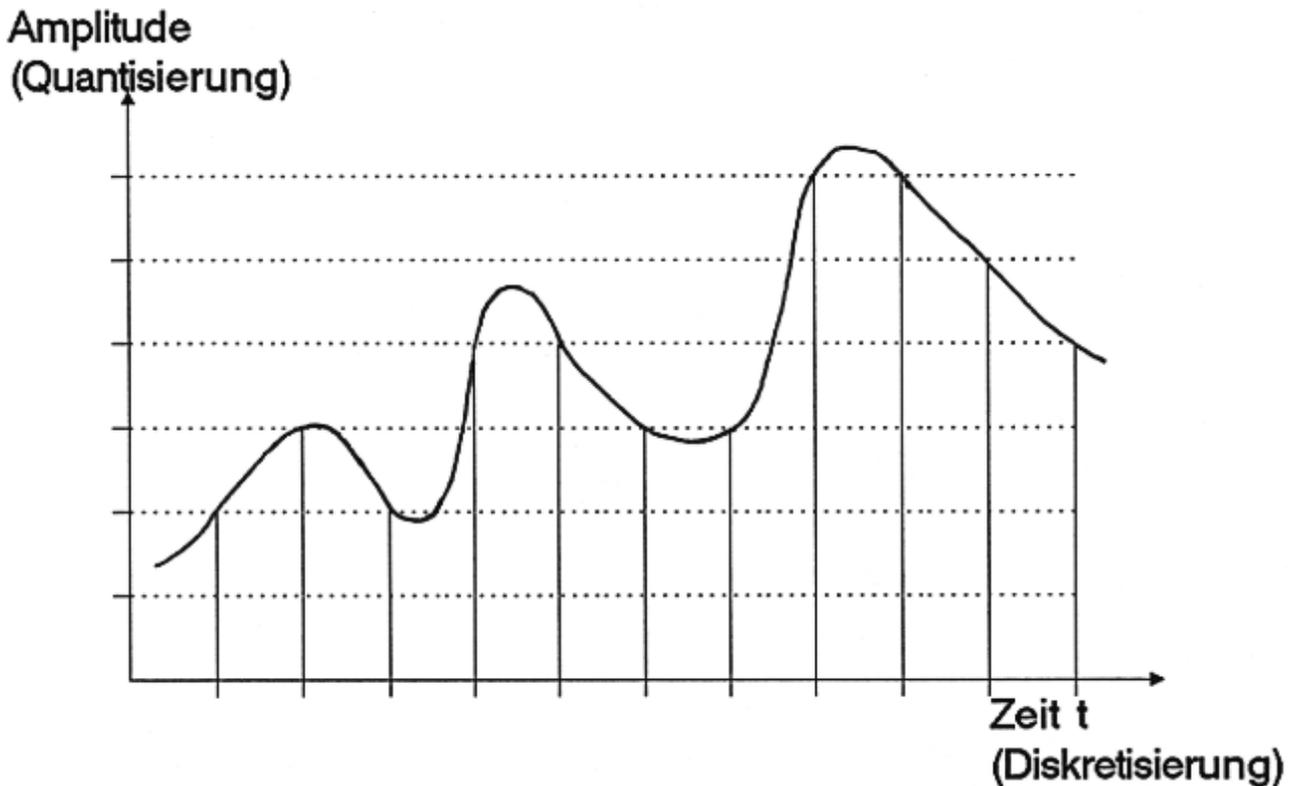


Abbildung 4.4: Bei einem analogen, zeitabhängigen Signal erfolgt die Diskretisierung durch Abtastung des Signals in festen Zeitabständen. Der gehörige analoge Signalwert wird einer beschränkten Menge von diskreten Signalwerten zugeordnet (Quantisierung).

4.4 Fehler bei der Bildaufnahme

Die Darstellung eines realen Bildes im Rechner geschieht prinzipiell in drei Schritten:

- **Abbildung auf den Sensor:** das Bild wird durch das optische System (Linsen) auf den Sensor abgebildet.
- **Ortsdigitalisierung:** es erfolgt eine Diskretisierung des Ortes.
- **Quantisierung:** die ortsdiskretisierten Intensitäten werden in Abhängigkeit der Quantisierungsstufen eingeteilt.

Erst nach Beendigung der letzten Stufe hat man ein im Rechner verwertbares "Zahlenbild" zur Verfügung. Leider haben alle an diesen Stufen beteiligten Bauteile und Verfahren Mängel, die sich in "falschen" Zahlenwerten im resultierenden Bild widerspiegeln.

Die folgende Tabelle zeigt die häufigsten Fehler und ihre Ursachen.

Fehlerquelle	Fehlerart
Optik	geometrische Verzerrung
	optische Dispersion
	Farbfehler
Sensor	Unschärfe
	Inhomogenität
	Farbfehler
	Übertragungsfunktion
	Unschärfe
A/D-Wandler	Aliasing
	Quantisierung
	Aliasing
Verstärker	Rauschen

Tabelle 4.1: Mögliche Fehlerquellen und Fehlerarten bei der Bildaufnahme.

Auch das Auge zeigt Abbildungsfehler (Bild wird auf dem Kopf aufgenommen; ortsabhängige Farb- und Hell/Dunkel-Empfindlichkeit; Verzerrungen usw.), die durch die Weiterverarbeitung noch im Auge oder in den einzelnen Teilen des Gehirns kompensiert und korrigiert werden.

Im nachfolgenden werden die einzelnen Fehlerquellen mit ihren möglichen Fehlerarten am Beispiel der CCD-Kamera näher untersucht. Für die anderen Bildaufnahmegeräte gilt ähnliches.

4.4.1 Geometrische Fehler

Das reale, zu erfassende Bild muß mit Hilfe geeigneter Transformationen (Skalierung, ggf. Spiegelung und Translation) auf die aktive Fläche des Sensors (wird als homogen angenommen) abgebildet werden. Dies geschieht über ein Linsen- oder Spiegelsystem. Da es bedingt durch Fertigungsverfahren und Materialeigenschaften keine perfekten Linsen gibt, treten bei der Abbildung Fehler auf. Am bekanntesten sind dabei die *geometrischen Verzerrungen*, die man bis zu einem gewissen Grad mit Hilfe anderer optischer Linsen wieder ausgleichen kann.

Verzerrungen können aber auch durch entsprechende geometrische Transformationen (Skalierung, Translation, Rotation usw.) beseitigt werden. Faktoren für die einzelnen Geräte sind anhand von Testbildern zu ermitteln. Die Transformationsgleichungen lauten (in homogenen Koordinaten):

4. Geräte zur Bildaufnahme

$$[x', y', 1] = [x, y, 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ D_x & D_y & 1 \end{bmatrix} \text{ Translation}$$

mit D_x, D_y als Translationsgröße in horizontaler bzw. vertikaler Richtung,

$$[x', y', 1] = [x, y, 1] \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ Skalierung}$$

mit S_x, S_y als Skalierungsfaktor in horizontaler bzw. vertikaler Richtung und

$$[x', y', 1] = [x, y, 1] \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ Rotation}$$

wobei α der Rotationswinkel in mathematisch positiver Richtung um den Ursprung ist [ES87].

4.4.2 Farb- und Intensitätsfehler

Optische Linsen sind nicht für alle spektralen Bereiche gleich durchlässig. Zusätzlich besitzt die lichtempfindliche Schicht einer Kamera noch unterschiedliche Empfindlichkeiten für die einzelnen Farben (daher gibt es unterschiedliche Einsatzgebiete von Kameraröhren, vergleiche Abschnitt 4.1). CCD-Sensoren sind im allgemeinen im roten und infraroten Bereich empfindlicher (wie die Unterschiede in der spektralen Empfindlichkeit des menschlichen Auges, vergleiche Abschnitt 3.2). Wird dies nicht berücksichtigt, so entstehen *Farbfehler* in Form von unrealistischen Abschwächungen bzw. Betonungen einzelner Farbtöne.

Mit Hilfe von Testbildern kann jedoch die Übertragungskurve einer Kamera für den gewünschten spektralen Bereich gemessen und durch entsprechende Gewichtungsfaktoren im Rechner korrigiert werden.

4.4.2 Diskretisierungs- und Quantisierungsfehler

Normalerweise wird die Dichte der lichtempfindlichen Elemente in der Kamera als homogen angenommen. Aus fertigungstechnischen Gründen muß dies jedoch nicht immer der Fall sein. Berücksichtigt man diese unterschiedlichen Bildpunkt-Dichten, so entstehen unterschiedliche Auflösungen (Schärfe) im Bild. Dies führt zu geometrischen Verzerrungen, wenn bei den folgenden Schritten die Punkt-Dichte als gleichmäßig angenommen wird.

Speziell beim CCD-Sensor kommt noch der Nachteil hinzu, daß bei den derzeit verfügbaren Elementen der lichtempfindliche Teil kleiner als die zu einem Bildpunkt gehörende Matrixfläche ist. Daraus resultiert eine Tiefpaßfilterung (siehe Abschnitt 10.5), d.h. Details zwischen den lichtempfindlichen Teilen werden nicht erfaßt. Dies kann teilweise durch eine Defokussierung (Unschärfe) im Linsensystem kompensiert werden. Der Verlust an Schärfe muß aber wieder herausgerechnet werden.

Übersteigen die Feinheiten und Details des Originalbildes das Auflösungsvermögen der Kamera, so entstehen durch die *Diskretisierung* Unterabtastungseffekte (*Aliasing*, vergleiche Abschnitt 5.52).

Unschärfen treten ebenfalls auf, wenn das Linsensystem (mit Blendenöffnung) nicht korrekt auf die Entfernung zum Objekt eingestellt ist. Auch durch die bei unterschiedlichen Wellenlängen variierende Brechung können Unschärfen bei nicht monochromen Objekten entstehen (*optische Dispersion*).

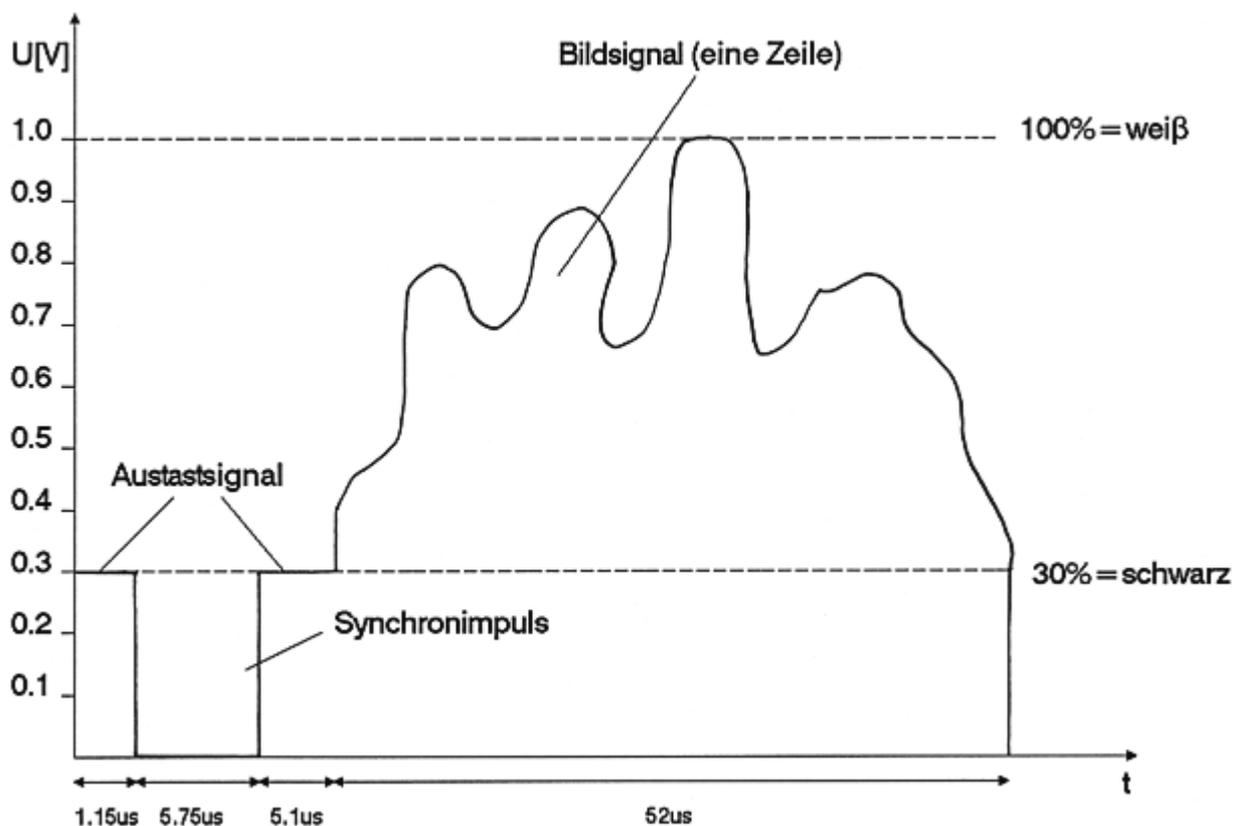


Abbildung 4.5: Prinzipieller Aufbau des Bild-Analogsignals (Videosignal). Die Ortsinformation steckt nur in der seit dem Synchronsignal vergangenen Zeit.

Ein weiterer Fehler bei der Bildwandlung in der Kamera ist bedingt durch die Querleitfähigkeit der Speicherelemente. Die Isolation zu den Nachbarpixeln ist nicht hoch genug, um ein "Zerfließen" der Ladung in die Umgebung zu verhindern. Dadurch entstehen Intensitätsverminderungen und Unschärfen.

4. Geräte zur Bildaufnahme

Die Umwandlung der unterschiedlichen Lichtintensitäten in elektrische Signale geschieht bedingt durch die verwendeten Bauteile nicht immer linear. Auch hier muß dies durch eine geeignete Übertragungsfunktion oder eine entsprechende *Quantisierung* wieder korrigiert werden.

Die Forderung, Bilder in Echtzeit, d.h. im Zeittakt der Aufnahme zu digitalisieren, bedingt eine hohe Verarbeitungsgeschwindigkeit in der Quantisierungsstufe. Je feiner ein analoges Signal quantisiert werden soll, desto länger dauert dieser Vorgang. Daher wurde bis vor wenigen Jahren die Quantisierung nur mit 4 Bit, entsprechend 16 Stufen betrieben. Heute sind jedoch bei normalen Video-Signalen bis zu 12 Bit Genauigkeit entsprechend 4096 Stufen möglich. Trotzdem können prinzipiell quantisierte Signale die analogen Werte nur annähern.

In der Kamera wird das zweidimensionale, diskrete Ladungsfeld in ein serielles Analogsignal umgewandelt. Dazu erfolgt eine pixel- und zeilenweise Abtastung des Bildes. Die einzelnen Punkte werden zeitlich hintereinander zum Verstärker gesendet. Die Information des Ortes steckt nunmehr ausschließlich im zeitlichen Abstand der Intensität zu einem Synchronimpuls. Eventuelle Störungen des Zeitgebers zerstören somit die Ortsinformation. Kurze Störungen im Signal verändern die Intensität des entsprechenden Pixels und führen zu *Rauschen*.

Ist es nicht notwendig, ein Bild in Echtzeit aufzunehmen, so können mehrere Bilder des gleichen Objektes/Szene aufgenommen und über die einzelnen Signale gemittelt werden. Im wesentlichen wird dabei der Signal-Rausch-Abstand erhöht und die Empfindlichkeit eventuell verbessert.

Aufgaben

Aufgabe 1

Eine CCD-Kamera mit einer Auflösung von 512×512 Bildpunkten nimmt eine Szene der Größe $30 \times 30 \text{ m}^2$ auf. Welche kleinsten Objekte können dabei theoretisch noch aufgelöst werden? Weshalb ist in der Praxis die Ortsauflösung geringer?

Aufgabe 2

Ein CCD-Chip mit einer Auflösung von 2048×2048 Pixel besitzt eine Chipfläche von $18.4 \times 18.4 \text{ mm}^2$. Jeder der quadratischen Bildpunkte besitzt eine Kantenlänge von $8 \mu\text{m}$. Wie groß ist die reale aktive Fläche im Verhältnis zur Gesamtfläche? Welche Auswirkung hat dies bei der Bildaufnahme?

Aufgabe 3

Erklären Sie den Unterschied zwischen

- optischer Dispersion und
- Farbfehlern.

Wodurch entstehen diese Fehler und wie kann man sie korrigieren?

Aufgabe 4

Ein Diapositiv der Größe $24 \times 36 \text{ mm}^2$ wird mit $25 \mu\text{m}$ Rasterflächengröße und drei Farbauszügen digitalisiert. Es ergibt sich ein Bild, bei dem sich jeder Bildpunkt aus den drei Farben Rot, Grün und Blau zusammensetzt. Die Intensität jeder einzelnen Farbe kann Werte von 0 bis 255 annehmen.

1. Wieviele Bildzeilen und Bildspalten hat das digitalisierte Bild?
2. Aus wieviel Bildpunkten besteht das Bild insgesamt?
3. Wieviele Bytes werden zur Speicherung eines Bildpunktes benötigt?
4. Wieviele Bytes werden zur Speicherung des gesamten Bildes benötigt?

4. Geräte zur Bildaufnahme

5. Geräte zur Bildwiedergabe

Die mit Hilfe von Kamera oder Scanner aufgenommenen Bilder sollten durch geeignete Wiedergabegeräte möglichst originalgetreu dargestellt werden. Eine Ausgabe kann über entsprechende Bildschirme oder auch Drucker erfolgen. Je nach Anwendungsgebiet haben diese unterschiedlichen Geräte mit ihren verschiedenen technischen Leistungsmerkmalen unterschiedliche Vor- und Nachteile. Der Aufbau der wichtigsten Geräte zur Bildwiedergabe sowie ihre Eigenschaften werden in den nachfolgenden Abschnitten erläutert.

5.1 Die Kathodenstrahlröhre

Die Elektronenstrahlröhre, auch *Kathodenstrahlröhre* oder englisch CRT (*Cathode Ray Tube*) genannt, ähnelt im Aufbau der Kameraröhre (siehe Abschnitt 4.1). Hier werden jedoch keine lichtempfindlichen Punkte mit einem Elektronenstrahl abgetastet, sondern der Elektronenstrahl schreibt das Bild auf eine Leuchtschicht. Zur Steuerung des Strahls werden elektronische Linsen in Form von Plattenkondensatoren oder Magnetspulen verwendet.

Eine Kathodenstrahlröhre zur monochromen Bildwiedergabe besteht aus den in Abbildung 5.1 dargestellten Komponenten. Die Vakuumröhre besitzt auf der einen Seite den eigentlichen Bildschirm, eine Mattscheibe, die mit einer speziellen Schicht einer Phosphorverbindung belegt ist. Diese Schicht hat die Eigenschaft, auftreffende Elektronen zu absorbieren und statt dessen Licht zu emittieren. Je nach Phosphorverbindung entstehen unterschiedliche Wellenlängen bzw. Lichtfarben. Die Helligkeit des ausgesandten Lichtes ist dabei proportional zur Intensität des Elektronenstrahls, jedoch nicht linear abhängig (vergleiche Abschnitt 5.5.1).

Auf der gegenüberliegenden Seite der Mattscheibe befindet sich die beheizte Kathode, um die sich während der Heizung eine Elektronenwolke bildet. Um daraus einen Elektronenstrahl zu erzeugen wird ein starkes elektrisches Feld benötigt. Dies wird durch Anlegen einer hohen Spannung zwischen Kathode und Anode erzeugt. Das elektrische Feld ist in der Lage, aus der Elektronenwolke um die beheizte Kathode Elektronen "herauszusaugen" und in Richtung auf den Bildschirm zu beschleunigen.

Einige Zusatzeinrichtungen sorgen dafür, daß der Elektronenstrahl gebündelt wird und man seine Intensität regeln kann. Das Bündeln (*Fokussierung*) geschieht durch Lochblenden, die Regelung der Intensität durch ein Steuergitter.

Zum Erzeugen eines Bildes auf dem Bildschirm muß dafür gesorgt werden, daß sich der Elektronenstrahl genügend schnell über jede Stelle des Bildschirmes bewegt. Diesen Vorgang nennt man *Ablenkung*, die dafür benötigte Vorrichtung *Ablenkteil*. Es gibt zwei verschiedene Prinzipien der Ablenkung:

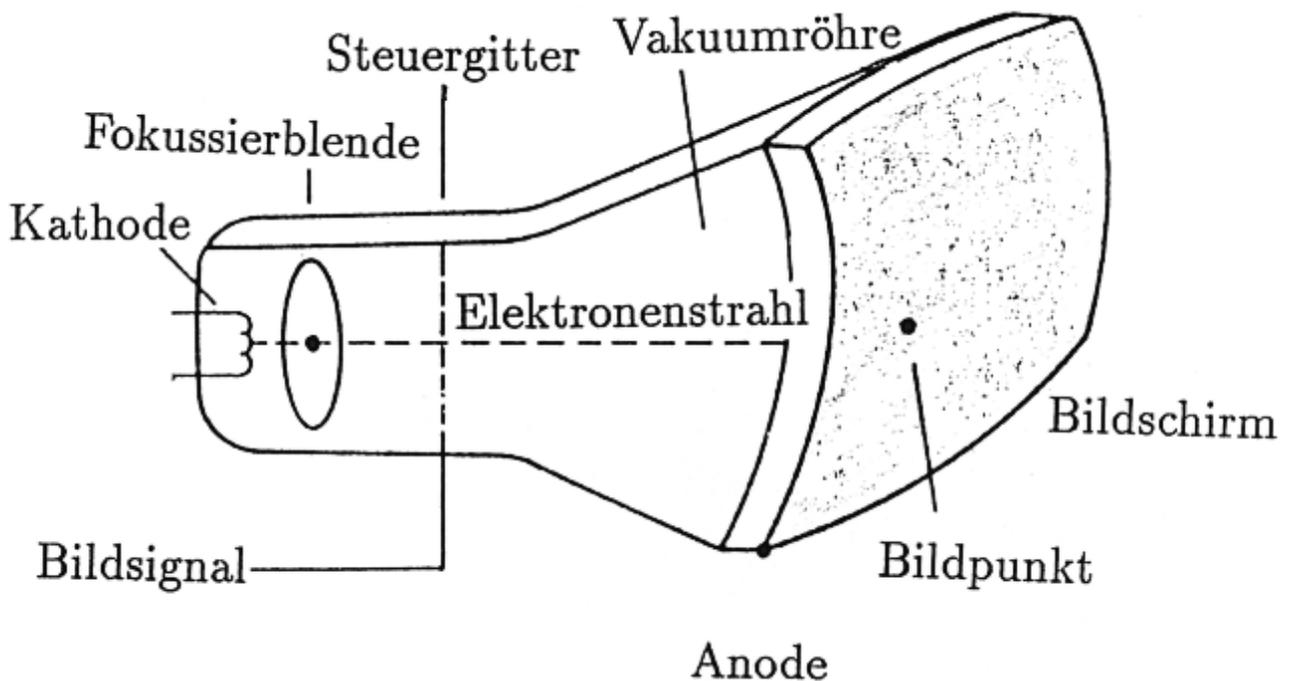


Abbildung 5.1: Schematisierter Aufbau der Kathodenstrahlröhre. Die Kathode emittiert Elektronen, die über entsprechende Einrichtungen zu einem Elektronenstrahl gebündelt werden. Dieser Strahl wird durch unterschiedliche Ablenkeinheiten auf die gewünschte Stelle des Bildschirms positioniert, wo je nach Intensität ein mehr oder minder heller Lichtpunkt entsteht.

Elektrostatische Ablenkung:

Der Elektronenstrahl wird durch einen Kondensator quer zur Strahlrichtung geschickt. Durch das elektrische Feld im Kondensator kann der Strahl abgelenkt werden; mit zwei um 90 Grad gedrehte Kondensatoren läßt sich jeder Bildpunkt erreichen. Wegen der geringen Ablenkleistung der Kondensatoren ergeben sich Probleme mit der Tiefe der Bildröhre bei größeren Bildschirmen. Allerdings arbeitet dieses Verfahren relativ genau und wird daher bei Meßgeräten (Oszillographen) eingesetzt.

Magnetische Ablenkung:

Bei diesem Verfahren wird der Elektronenstrahl durch Magnetfelder abgelenkt. Dabei werden zwei um 90 Grad gedrehte Elektromagnete (Spulen) verwendet. Vorteil dieses Verfahrens ist die große Ablenkleistung. Große Ablenkleistung bedeutet einen großen Ablenkwinkel und ermöglicht damit den Bau relativ kompakter Bildröhren. Die Positioniergenauigkeit ist aber nicht so hoch wie bei der elektrostatischen Ablenkung.

5.1.1 Der Phosphor

Die Innenwand des Bildschirms ist mit einer Phosphorverbindung belegt, die beim Auftreffen des Elektronenstrahls Licht emittiert (*Fluoreszenz*). Nach dem Abschalten des Elektronenstrahls wird einige Zeit weiter Licht mit abnehmender Helligkeit ausgesendet (*Phosphoreszenz*). Die Nachleuchtdauer (*Persistenz*) ist bei der Auswahl einer Bildröhre neben der Farbe und der Helligkeit von größter Wichtigkeit. Sie bestimmt, wie oft das Bild regeneriert, d.h. wiederholt erzeugt werden muß, damit der Benutzer den Eindruck eines stehenden, flimmerfreien Bildes gewinnt. Bei schnell bewegten Bildern mit hoher Bildwiederholrate darf die Persistenz nicht zu groß sein, da ansonsten noch die Schemen voriger Bilder auf dem Bildschirm neben dem aktuellen Bild sichtbar sind [ES87].

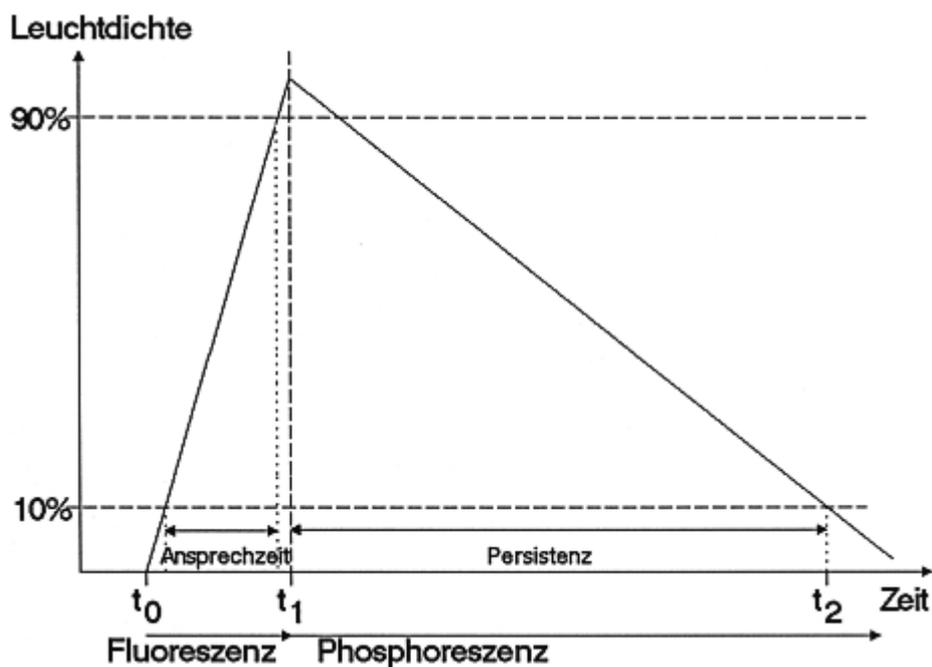


Abbildung 5.2: Die Lichtemission in Abhängigkeit von der Zeit. Zum Zeitpunkt t_0 wird der Elektronenstrahl an der Bildschirmposition eingeschaltet und zum Zeitpunkt t_1 wieder ausgeschaltet.

Aufgrund der unterschiedlichen Persistenz der einzelnen Phosphorverbindungen ergeben sich unterschiedliche Einsatzgebiete.

5.1.2 Vektorsichtgerät

Bisher wurde nur der prinzipielle Aufbau der Kathodenstrahlröhre beschrieben, nicht jedoch der Vorgang der Bilderzeugung auf der Mattscheibe. Es werden hier zwei Arten von Sichtgeräten unterschieden: das Vektorsichtgerät und das Rastervisgerät.

5. Geräte zur Bildwiedergabe

Beim *Vektorsichtgerät* besteht das zu erzeugende Bild aus einer Vielzahl von Vektoren. Die Anfangs- und Endkoordinaten dieser Vektoren sind im Bildspeicher abgelegt. Aufgrund dieser Daten werden dann die Vektoren auf die Mattscheibe geschrieben.

Phosphor	Fluoreszenz Phosphoreszenz	Persistenz	Anwendung
P ₁	gelblich-grün	22 msec	Monitor u.ä.
P ₄	weiß	20 µsec blaue Komp. 60 µsec gelbe Komp.	Fernsehen
P ₇	gelb-grün	50 µsec blaue Komp. 350 µsec gelbe Komp.	Radar
P ₁₂	orange	200 msec	Radar
P ₁₆	bläulich-violett	0.12 µsec	Flying Spot

Tabelle 5.1: Die Leuchtfarbe und die Persistenz einiger wichtiger Phosphorverbindungen.

Der Elektronenstrahl ist bei Vektorsichtgeräten frei positionierbar. Ist die Liste der darzustellenden Vektoren abgearbeitet, so kann das Bild neu aufgebaut werden. Die Bildwiederholrate ist somit abhängig von der Anzahl zu zeichnender Vektoren und deren Länge, also von der Komplexität des Bildes. Bei sehr umfangreichen Bildern kann die Bildwiederholrate soweit absinken, daß das dargestellte Bild deutlich flimmert oder sogar flackert.

Zur Darstellung von einigen wenigen Farben werden bei Vektorsichtgeräten sogenannte *Penetration-Röhren* verwendet [SS87]. Auf der Mattscheibe sind hier zwei Schichten von unterschiedlichen Phosphorverbindungen (üblicherweise Rot und Grün) aufgebracht. Je nach Energie des auftreffenden Elektronenstrahls wird entweder nur die rote Schicht (ca. 6 kV), die grüne Schicht (ca. 12 kV) angeregt oder es entsteht eine Mischfarbe (ca. 9 kV). Die Umschaltzeit zwischen den einzelnen Farben ist relativ lang, so daß der Umfang der flimmerfrei darstellbaren Bildinformation nicht durch die Komplexität des Bildes, sondern durch diese Umschaltzeit bestimmt wird.

5.1.3 Rastersichtgerät

Die *Rastersichtgeräte* sind die heutzutage am häufigsten verwendeten Sichtgeräte. Die darzustellende Bildinformation wird dazu in einzelne Bildpunkte, Zeilen und Spalten aufgeteilt. Die Speicherung der Bildpunkte geschieht im Bildschirmspeicher, einer Matrix mit einer Speicherposition für jeden darzustellenden Bildpunkt.

5.5 Fehler bei der Bildwiedergabe und deren Beseitigung

Der Elektronenstrahl der Kathodenstrahlröhre beschreibt in einem festen Zeitschema die Mattscheibe zeilenweise von links nach rechts und von oben nach unten. An den einzelnen Stellen wird der korrespondierende Bildpunkt aus dem Bildschirmspeicher ausgelesen und der Elektronenstrahl in seiner Intensität nach dem Helligkeitswert des Bildpunktes geregelt.

Die Bildwiederholfrequenz ist somit unabhängig von der Komplexität des Bildes und nur durch technische Faktoren, wie die Zeit zum Auslesen eines Bildpunktes aus dem Bildschirmspeicher sowie der Zeit zur Positionierung und Regelung des Elektronenstrahls, begrenzt. Die Auflösung wird durch die Auflösung des Bildschirmspeichers bestimmt.

Damit die Auswirkung des Elektronenstrahls lokal eng begrenzt bleibt und es zu (fast) keinem Übersprechen zwischen benachbarten Bildpunkten kommt, wird der Durchmesser des Strahls durch eine sogenannte Lochmaske begrenzt (siehe Abbildung 5.3).

Zur farbigen Darstellung werden drei Elektronenstrahlssysteme verwendet, die unterschiedliche Phosphorleuchtpunkte oder -streifen in den drei Grundfarben ansprechen. Der Bildspeicher ist entweder auch in drei Ebenen aufgeteilt oder die Intensitätsinformation der Grundfarben wird über Farbtabelle bestimmt.

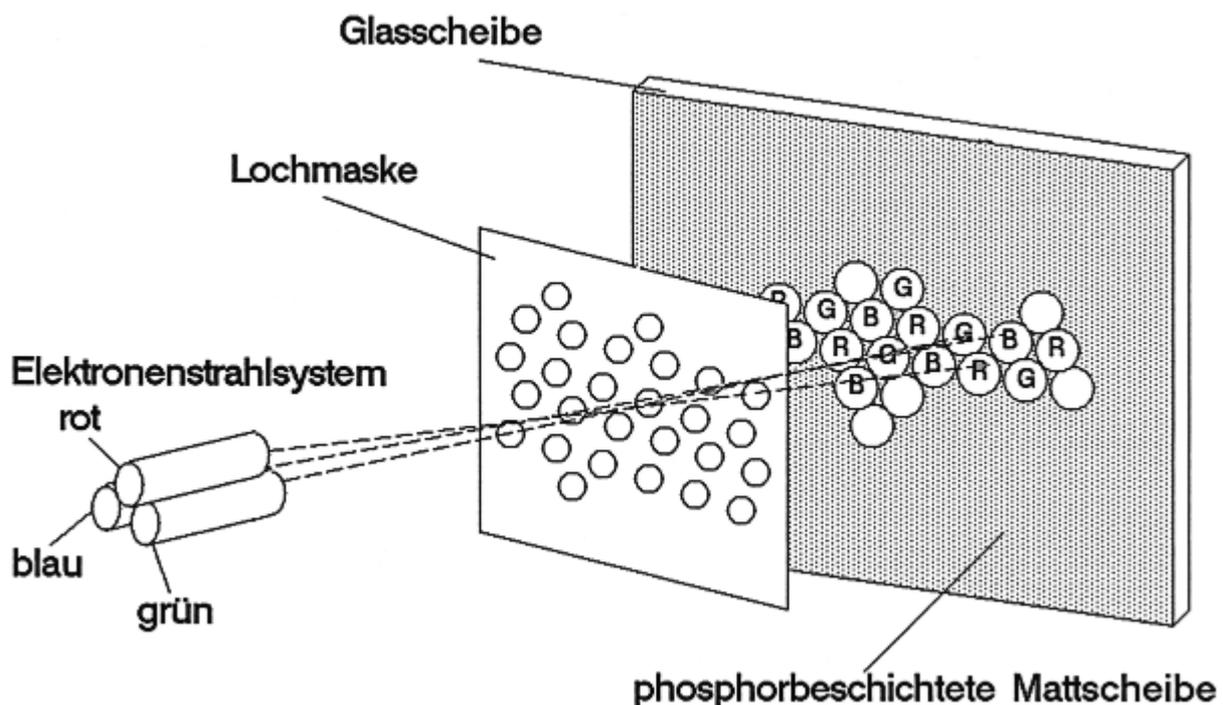


Abbildung 5.3: Aufbau einer Lochmaskenröhre zur Farbdarstellung mit Delta-Anordnung der Elektronenstrahlssysteme und der Farbpunkte.

Bei der Lochmaskenröhre mit *Delta-Anordnung* sind sowohl die drei Elektronenstrahlssysteme für Rot, Grün und Blau als auch die Phosphorfarbpunkte deltaförmig angeordnet. Je drei benachbarte Punkte auf der Leuchtschicht mit den Grundfarben ergeben einen farbigen Bildpunkt. Eine Lochmaske wird verwendet, um sicherzustellen, daß jeder Strahl nur den für ihn vorgesehenen Farbpunkt anspricht. Die Strahlen für die roten, grünen und blauen Punkte müssen die Maskenöff-

5. Geräte zur Bildwiedergabe

nung im richtigen Winkel passieren, um ihre zugehörigen Phosphorpunkte zu treffen (*Konvergenz*). Bei entsprechendem Betrachtungsabstand verschmelzen die drei einzelnen monochromen Punkte für den Betrachter zu einem einzigen Farbpunkt. Der Mindestabstand berechnet sich aus dem Verhältnis von Sehwinkel zur maximalen Sehschärfe. Der Sehwinkel beträgt bei ruhendem Auge ca. 10 Bogengrad, die maximale Sehschärfe ca. 1' (Bogenminute), woraus sich ein Verhältnis von 600 : 1 ergibt. Für eine Auflösung von 600 Zeilen ist somit ein Betrachtungsabstand von etwa der 6fachen Bildhöhe sinnvoll, da das Auge dann die einzelnen Zeilen und Punkte noch nicht getrennt wahrnehmen kann.

Bei der *Inline-Anordnung* werden, wie der Name schon sagt, die Elektronenstrahlsysteme in einer Zeile angeordnet. Daraus ergibt sich, daß die Farbtripel nicht aus drei Punkten, sondern aus drei Zeilen bestehen (siehe Abbildung 5.4). Es wird auch keine Lochmaske, sondern eine Schlitzmaske verwendet. Damit die drei Farbstreifen für den Betrachter zu einem Farbpunkt verschmelzen, müssen diese entsprechend dünn sein und eng beieinander liegen. Der Vorteil dieses Systems liegt in der einfacheren Ablenkung und der besseren Konvergenz.

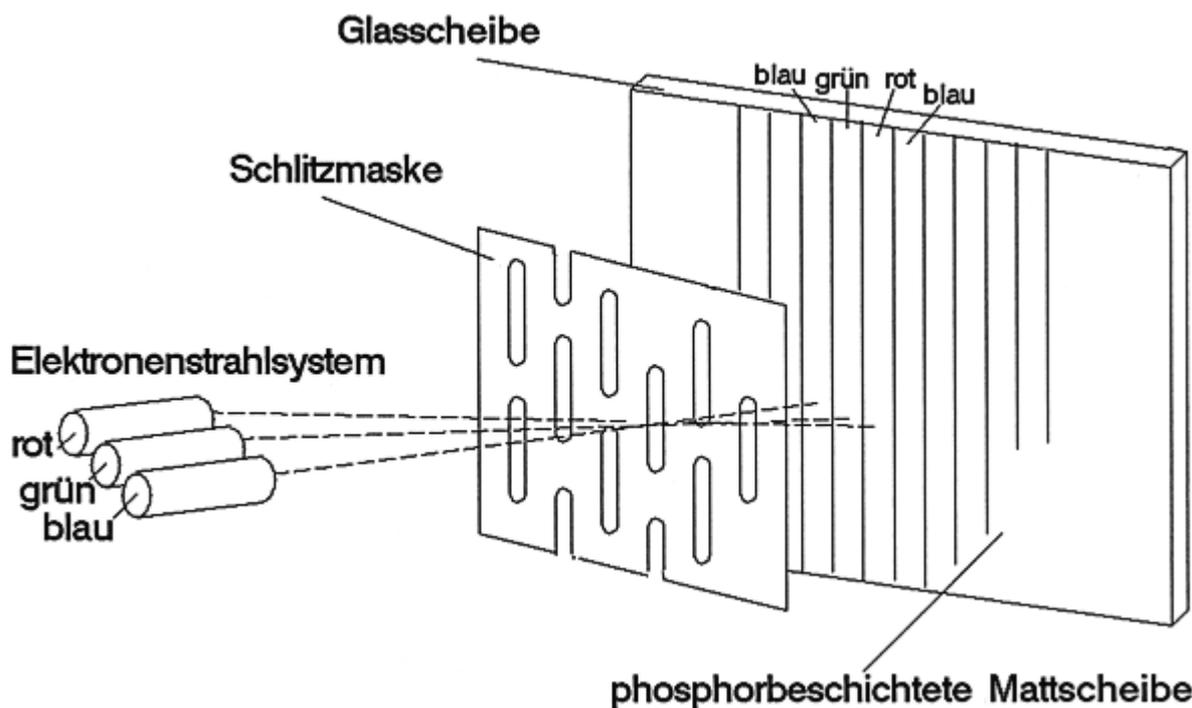


Abbildung 5.4: Bei der Schlitzmaskenröhre mit Inline-Anordnung wird ein Farbpunkt aus drei nebeneinander liegenden Farbstreifen gebildet.

Die maximale Auflösung von Rastersichtgeräten ist durch die Geschwindigkeit, mit der die einzelnen Bildpunkte auszulesen sind, und durch die Bandbreite des Systems beschränkt. Auflösungen von Farbgeräten mit 1280×1024 Bildpunkten sind Standard. Geräte mit über 2048×1536 Bildpunkten sind ebenfalls schon erhältlich. Bei Schwarzweißmonitoren sind Ausfertigungen in 19 Zoll mit 4096×3300 Bildpunkten erhältlich. Das entspricht mit einer Punktdichte von 300 dpi (Dots per Inch) genau der Auflösung gängiger Laserdrucker.

5.2 Das LC-Display

Das Prinzip der Flüssigkristall-Anzeige (*LCD=Liquid-Crystal-Display*) beruht auf der Streuung des Lichtes, d.h. passive Effekte spielen hierbei eine Rolle, im Gegensatz zur Kathodenstrahlröhre, die Licht emittiert. Die Streuung des Lichtes ist bei den LCD-Zellen abhängig von der Höhe der angelegten Spannung, wodurch sich unterschiedliche Graustufen realisieren lassen.

Ein LC-Display besteht im Prinzip aus zwei parallelen, sandwichartigen Glasplatten, die sich im Abstand von weniger als $100\mu\text{m}$ voneinander befinden. Die Innenseiten dieser Platten sind mit einem Muster transparenter Elektroden versehen, während an der Außenseite zwei rechtwinklig zueinander orientierte Polarisationsfilter angebracht sind. Der Zwischenraum zwischen den Glasplatten ist mit einer Flüssigkeit gefüllt, die unter anderem aus Verbindungen wie aromatische Ester oder Schiffsche Basen besteht. In bestimmten Temperaturbereichen tritt bei diesen Verbindungen das Phänomen auf, daß sich fadenförmige (nematische) Moleküle bilden, die in ihren optischen Eigenschaften Kristallen gleichen. Daher die Bezeichnung Flüssigkristall. Die Moleküle können sich zwischen den Glasplatten frei bewegen. Diese Eigenschaft der Flüssigkristalle wurde Ende der 60er Jahre entdeckt. Befindet sich ein solches Flüssigkristall in einem elektrischen Feld, so wirkt auf die Moleküle wegen der asymmetrischen Ladungsverteilung ein Drehmoment. Das hat zur Folge, daß sich die Moleküle bei ausreichend hoher Feldstärke parallel zur Feldrichtung orientieren. Dazu ist jedem "Bildpunkt" ein Speicherelement zugeordnet. Als Speicherelement wird ein Kondensator benutzt, der über einen entsprechend angesteuerten MOS-Transistor aufgeladen wird. Eine derartige Anordnung ist erforderlich, da die Zustandsänderungen der Flüssigkristalle wegen der hohen Viskosität der organischen Flüssigkeit nur sehr träge erfolgt, die Spannung also für eine gewisse Zeit an der Flüssigkristall-Zelle anliegen muß.

Legt man nun ein äußeres Feld an, orientieren sich die Moleküle so, daß sie die Polarisationsachse des durchgehenden Lichtes drehen, so daß sich im Zusammenwirken mit den beiden rechtwinklig zueinander orientierten Polarisationsfiltern entsprechend der Form der Elektroden dunkle Bildzonen durch Ausfiltern allen Lichtes schaffen lassen. Zur Verbesserung des Kontrastes wird nicht das von außen auftreffende Licht für die Reflexion benutzt, sondern es wird eine Hintergrundbeleuchtung, eine Lichtquelle, verwendet. Um mit einer möglichst geringen elektrischen Spannung eine möglichst proportional große Richtungsänderung der Moleküle und damit wiederum einen höheren Kontrast zu erreichen, werden die Moleküle mechanisch unter Spannung gesetzt. Man spricht dann von "Supertwisted LCD".

Der große Vorteil der LC-Bildschirme ist die flache Bauweise und der niedrige Stromverbrauch (ca. 0.15 mW pro cm^2). Daher eignen sie sich besonders für tragbare Rechner und für Anzeigen, für die nur wenig Platz vorhanden ist. Ein Nachteil für die Bewegtbilddarstellung ist die relativ lange Erholzeit (Umschaltzeit) dieser nematischen Zellen, die bis zu 50 ms betragen kann. Da zu jedem Bildpunkt ein aktives Element in Form eines Transistors gehört, muß die Qualität der LC-Bildschirme sehr groß sein. Ein Ausfall eines Transistors führt zum Ausfall eines Bildpunktes, ein sichtbarer Defekt. Negativ fällt auch die geringe Leuchtdichte von etwa $5 - 10\text{ cd/m}^2$ ins Gewicht.

5. Geräte zur Bildwiedergabe

Bei der Kathodenstrahlröhre werden normalerweise Werte von 300 cd/m^2 mit Spitzenwerten bis zu 1500 cd/m^2 erreicht.

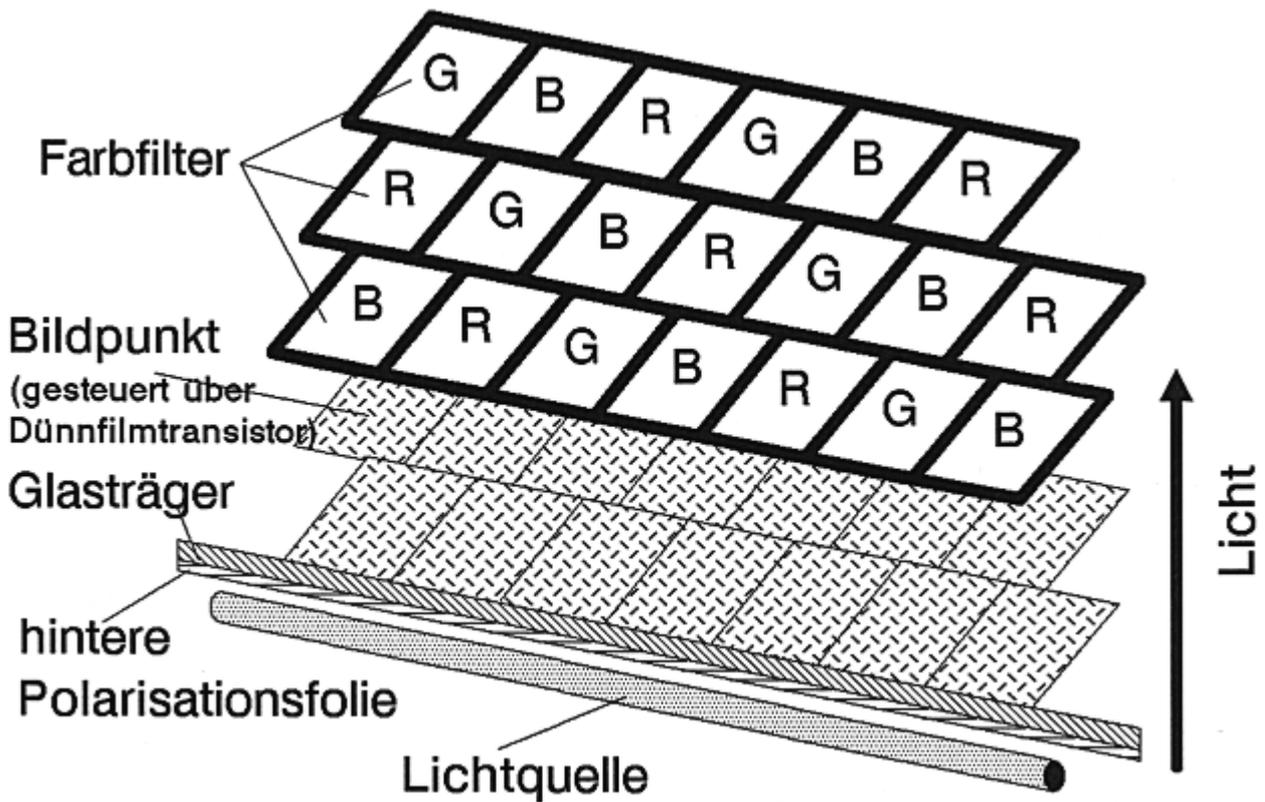


Abbildung 5.5: Querschnitt durch einen Farb-LC-Bildschirm. Die Abstände und Größen der einzelnen Schichten sind zur besseren Darstellung übertrieben dargestellt. Je drei Punkte in den Grundfarben Rot, Grün und Blau bilden einen Farb-LCD-Bildpunkt.

Im Farbdisplay werden für jeden darzustellenden Farbpunkt drei auf der Rückseite des Displays aufbrachte Transistoren für das Farbtupel Rot/Grün/Blau benötigt. Die eigentliche Farbe entsteht durch jeweils drei Filter in den Grundfarben. Dabei bilden die Grundfarben miteinander jeweils ein Dreieck. Mit 0.2 mm bis 0.4 mm liegt der Bildpunktabstand in der Größe der Kathodenstrahlröhre.

5.3 Das Plasma-Display

Ein *Plasma-Display* besteht aus zwei durchsichtigen und gasdichten Platten, die in einem geringen Abstand voneinander parallel angeordnet und deren Ränder gasdicht verbunden sind. Zwischen den Platten befindet sich eine gläserne Lochmaske, die den Raum in ein Raster von kleinen Zellen (Zylinder) aufteilt. Diese sind mit einem Gas (vorzugsweise Neon und Argon) gefüllt. Der Gasdruck liegt deutlich unter Atmosphärendruck. Jede der Platten trägt an der Seite zum Gasraum parallele Leiterbahnen (Kathode und Anode) in Übereinstimmung mit der Rastermaske und ist so angeordnet, daß die Leiterbahnen rechtwinklig zueinander stehen. Kleine Zuleitungen führen von den Leiterbahnen zu den einzelnen Entladungszellen (siehe Abbildung 5.6).

Wird eine Zündspannung in geeigneter Höhe zwischen Kathode und Anode angelegt, so wird das Gas (bei monochromen Displays Neon) in der dazwischen liegenden Zelle ionisiert (es entsteht ein *Plasma*) und beginnt zu leuchten. Nach dem Zünden einer Zelle kann die angelegte Spannung von der Zündspannung auf die wesentlich niedrigere Brennspannung verringert werden. Das Plasma bleibt dabei weiter bis zur Erreichung der Löschspannung bestehen.

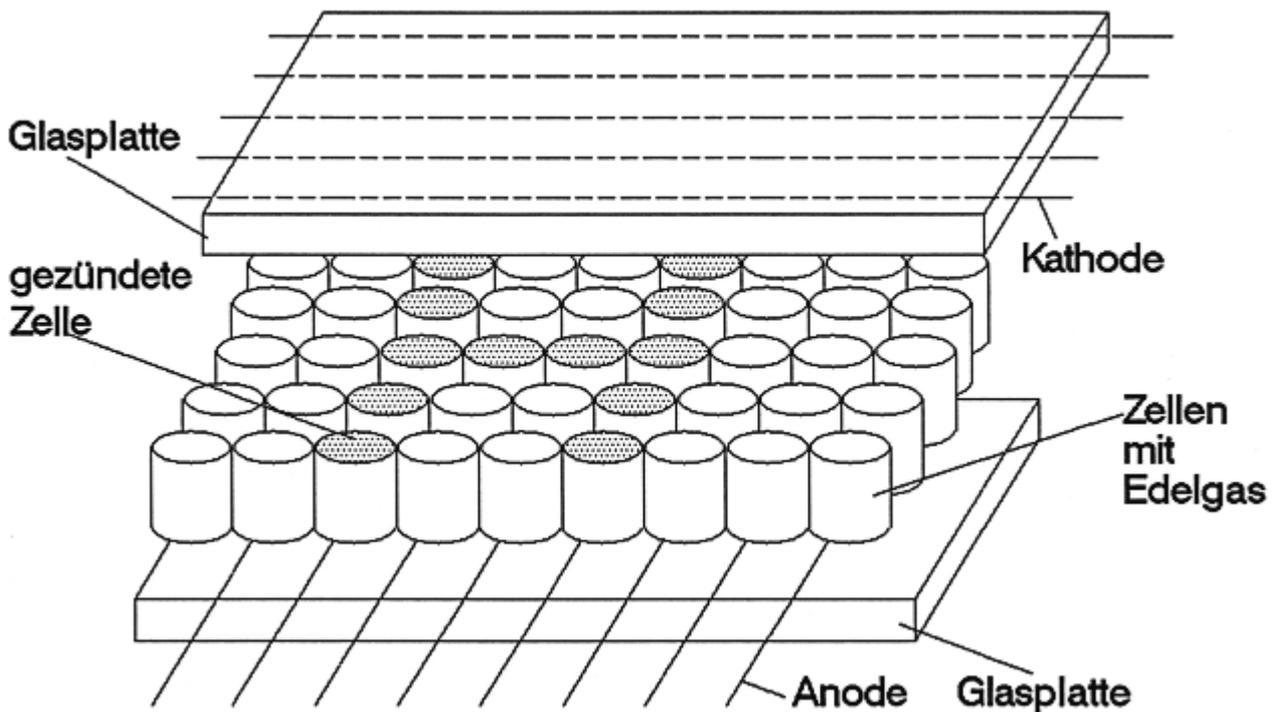


Abbildung 5.6: Der sandwichartige Aufbau des Plasma-Displays. In den einzelnen Zellen befindet sich ein Edelgas, das nach dem Anlegen einer relativ hohen Zündspannung ionisiert wird. Dieses entstehende Plasma sendet Licht entweder im sichtbaren oder im ultravioletten Bereich aus. Das UV-Licht wird durch eine Leuchtschicht an den Wänden der Zellen in sichtbares Licht umgewandelt.

Für andersfarbige Displays findet ein Gas Verwendung, das zu ultravioletter Lichtstrahlung (UV-Licht) angeregt wird. Das UV-Licht trifft auf den phosphoreszierenden Leuchtstoff, der sich an der Innenwand der Zylinder befindet. Dadurch wird der Leuchtstoff angeregt und wandelt unsichtbares UV-Licht in sichtbares Licht um.

Die Farbe des Lichtes hängt von der chemischen Zusammensetzung des Leuchtstoffes ab (vergleiche Abschnitt 5.1). Farb-Plasma-Displays sind somit auch realisierbar. Leider kann man nicht den gleichen Leuchtstoff wie bei der Kathodenstrahlröhre verwenden. Auch ist es momentan noch nicht möglich, ein Blau mit entsprechender Leuchtkraft zu erzeugen.

Neben der flachen Bauweise und dem flimmerfreien Bild sind der wesentlich bessere Kontrast im Vergleich zu den LC-Bildschirmen sowie die höhere Leuchtdichte von bis zu 150 cd/m^2 die großen Vorteile des Plasma-Displays. Die Bildpunktabstände liegen mit 0.3 mm bis 0.4 mm in den

5. Geräte zur Bildwiedergabe

gewohnten Größen. Negativ machen sich die hohe Spannung zum Erzeugen des Plasmas, der hohe Stromverbrauch und die relativ lange Abschaltzeit angesteuerter Bildpunkte bemerkbar.

Durch Ausnutzung des "Townsend-Effektes" sind Leuchtdichten bis zu 680 cd/m^2 und höhere Schaltgeschwindigkeiten realisierbar. Der irische Physiker Townsend entdeckte Anfang dieses Jahrhunderts, daß die Intensität der UV-Strahlung bei einer Gasentladung etwa 200 psec nach dem Anschalten ein Maximum durchläuft. Sie fällt dann nach etwa 500 psec auf die stabilere Glimmentladung mit wesentlich geringerer Intensität ab. Diese Glimmentladung wird bei den bisherigen Plasma-Displays verwendet. Nutzt man den Townsend-Effekt aus und steuert die einzelnen Bildpunkte bis zu 100000 Mal pro Sekunde an, so führt dies zu einer bis zu 20fach höheren Lichtausbeute.

5.4 Sonstige Wiedergabegeräte

Die oben vorgestellten drei Bildschirmarten repräsentieren in ihrer Reihenfolge die Bedeutung für den täglichen Einsatz. In Spezialgebieten gibt es noch weitere Displayarten. Einige interessante Entwicklungen sollen der Vollständigkeit halber hier noch einmal angesprochen werden.

Projektionsdisplays

Das *Projektionsdisplay* besteht aus einem durchsichtigen LC-Display. Es können dabei bis zu 2048×2048 Bildpunkte aufgelöst werden. Ein schneller Laserstrahl schreibt auf dem Flüssigkristalldisplay das Bild, welches anschließend von einer Xenonlampe durch die Flüssigkristalle hindurch über Linsen und Spiegel entsprechend vergrößert auf eine Projektionswand geworfen wird. Ein Vergrößern/Verkleinern des Bildes ist aufgrund der Linsen und Spiegel einfach möglich. Der Vorteil eines solches Displays ist neben der Größe auch der gute Kontrast.

Ferroelektrisches LC-Display

Eine interessante Besonderheit unter den Flüssigkristallanzeigen sind die *ferroelektrischen LCDs*. Sie unterscheiden sich von normalen Displays neben der höheren Schaltgeschwindigkeit und dem höheren Kontrast vor allem durch eine automatische Speicherfähigkeit. Einmal eingeschriebene Daten können ohne angelegte Hilfsspannung "eingefroren" werden. Leider ist die Speicherung nicht gegen mechanische Erschütterungen resistent. Durch solche Störungen kann die Anzeige gelöscht werden. Bisher gibt es auch nur ferroelektrische Schwarzweißdisplays.

Elektrolumineszenz-Display

Das Elektrolumineszenz-Display beruht auf dem 1936 von Georges Destriau entdeckten Effekt, daß bestimmte Verbindungen bei Anlegen eines elektrischen Feldes Licht emittieren. Die Treiberspannung muß je nach Verbindung zwischen 70 und 90 Volt oder 240 und 400 Volt liegen. Das Pixelraster bei diesen Bildschirmen wird durch eine netzförmige transparente Frontelektrode und einen Zinksulfid-Leuchtschirm gebildet. Damit wird eine Auflösung bis zu 1280×1280 Pixel und ein viermal höherer Kontrast als bei normalen LCDs erreicht. Die Auflösung ist mit denen der anderen Displayarten vergleichbar.

Bisher sind nur Displays mit zwei Intensitätsstufen verfügbar. Graustufen werden daher mit unterschiedlichen Punktdichten oder Halbtonverfahren simuliert (vergleiche Kapitel 6). Erste Versuche mit Farbdisplays sind mit einer streifenförmigen Anordnung der Farben gemacht worden. Jedoch fehlt die Farbe Blau noch fast vollständig, das RGB-Verhältnis stimmt nicht.

5.5 Fehler bei der Bildwiedergabe und deren Beseitigung

Einige der bei der Bildaufnahme beobachteten Fehler sind auch bei der Bildwiedergabe auf dem Monitor zu bemerken. Bedingt durch die physikalischen Grenzen der Ablenkeinheiten bei Kathodenstrahlröhren entstehen zum Bildrand hin Verzerrungen, die durch Vergleich mit verschiedenen Testbildern ermittelt und mit Hilfe entsprechender Schaltungen zum Teil kompensiert werden können.

Auch die falsche Wiedergabe von Farben (mangelnde Farbtreue) ist häufig anzutreffen. Durch unterschiedliche Gewichtung der einzelnen Farbkomponenten oder über Farbtabellen (Look-Up-Tabellen, LUT) sind diese Fehler korrigierbar. Jedoch spielt der subjektive Eindruck des einzelnen Betrachters hier eine große Rolle.

5.5.1 Helligkeitsfehler – Die Gamma-Korrektur

Wie schon in Abschnitt 5.1 beschrieben, ist die ausgesendete Lichtmenge bei Kathodenstrahlröhren zwar proportional jedoch nicht linear zur Intensität des Elektronenstrahls. Mit den normalerweise am Bildschirm angebrachten Reglern für Helligkeit (Faktor α) und Kontrast (Faktor γ) kann die Intensität in Abhängigkeit der Spannung V (normalisiert auf den Bereich 0–1) des Elektronenstrahls in gewissen Grenzen geregelt werden [Bli89a].

$$\text{Intensität} = \alpha V^\gamma$$

Es ist weiterhin zu berücksichtigen, daß der menschliche Gesichtssinn relative und nicht absolute Helligkeiten bewertet (siehe Kapitel 3). Die subjektive Helligkeitsfunktion hat einen logarithmischen Verlauf. Deshalb müssen die Grauwerte bzw. Farbwerte zwischen Schwarz und Weiß loga-

5. Geräte zur Bildwiedergabe

rithmisch gestuft sein. Bei der Darstellung auf dem Sichtgerät müssen zusätzlich die Nichtlinearitäten des eingesetzten Leuchtstoffs kompensiert werden. In vielen Systemen werden beide Effekte mit der sogenannten *Gamma-Korrektur* (γ -Korrektur) weitgehend beseitigt. Dazu wird in Meßreihen die Kennlinie des Leuchtstoffs bestimmt und bei der Darstellung von Objekten über eine Grau-/Farbwerttafel (*Look-Up-Table*, *LUT*) kompensiert. Die Nichtlinearität des Monitors wird in der Größe γ angegeben. Der Wert liegt im allgemeinen im Bereich von 2.0 bis 3.0. Die LUT berechnet sich nach der Formel (Intensität und LUT normiert auf den Bereich 0-255):

$$\text{LUT-Eintrag} = \text{INT}(((\text{Intensität}/255.0)^{1.0/\gamma}) * 255.0 + 0.5)$$

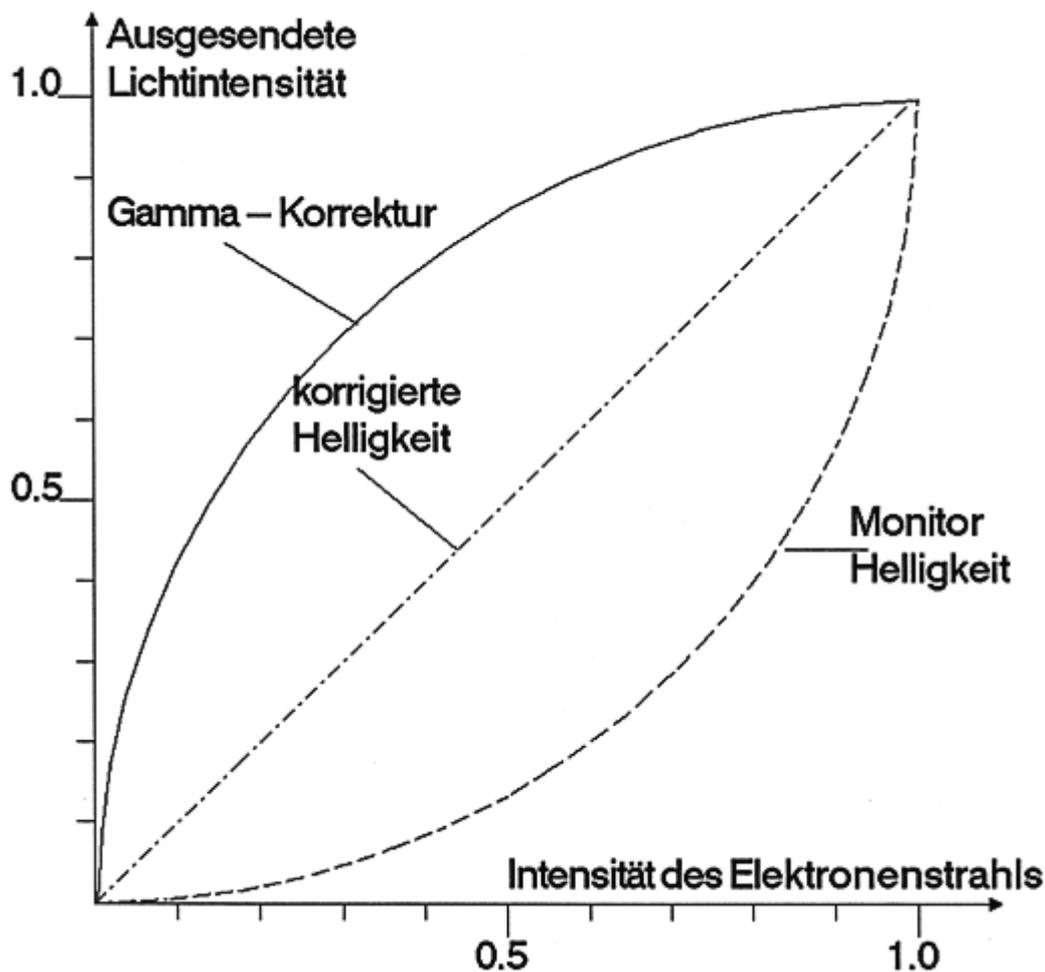


Abbildung 5.7: Die Gamma-Korrektur kompensiert die Nichtlinearität des eingesetzten Leuchtstoffs. Geringere Intensitäten werden daher stärker angehoben, was insgesamt zu einem Verlust von darstellbaren Intensitätsstufen führen kann.

Nach erfolgter Gamma-Korrektur mit dem richtigen Faktor γ wirken die betrachteten Bilder in ihrer Helligkeit und im Kontrast natürlicher. Jedoch sollte man beachten, daß nach dieser LUT-Transformation Graustufen verlorengegangen sind! Es sind weniger unterschiedliche Grautöne vorhanden, Schwarz wird aber noch auf Schwarz, Weiß auf Weiß abgebildet.

5.5 Fehler bei der Bildwiedergabe und deren Beseitigung

(Anmerkung: Das nachfolgende kleine Programm soll zur Einführung dienen. Der Aufbau entspricht dem in den später verwendeten Programmen. Den Prozeduren und Funktionen wird dabei die Nummer des Originalbildes `BildNr` und ggf. des Zielbildes übergeben. Die einzelnen Bilder sind in der Struktur `Picture[BildNr]` abgelegt. Diese ist im Anhang B ausführlich beschrieben. Die Prozeduren `ClearTextWindow` und `WriteText` dienen zum Löschen eines Bildschirmbereiches bzw. zur Ausgabe eines Textes an einer gewünschten Stelle. Über die Funktion `Read_Float` kann fehlertolerant eine Gleitkommazahl mit der Vorgabe eines Default-Wertes eingelesen werden.)

```
void Gamma_Korrektur(int BildNr)
/* BildNr = Nummer des zu bearbeitenden Bildes */
{
    int z,s; /* Laufvariablen fuer Zeilen/Spalten */
    unsigned char Grauwert[MAX_COLOR]; /* LUT fuer die Gamma-Korrektur*/
    float Gamma; /* Einzulesender Gamma-Wert */

    ClearTextWindow(35,18,80,25);
    WriteText(35,18,"* Gamma-Korrektur *");

    /* Gamma-Wert einlesen und Wert 2.3 als Default-Wert vorgeben */
    Gamma=Read_Float("Welchen Wert fuer Gamma", 2.3, ':');

    /* Neue, geaenderte LUT ueber die Gamma-Korrektur berechnen */
    for (i=0; i<MAX_COLOR; i++)
        Grauwert[i] = int( pow(double(i), double (1.0/Gamma) );

    /* Grauwerte transformieren und im Ausgangsbild speichern */
    for (z=0; z<Picture[BildNr].Zeilen; z++)
        for (s=0; s<Picture[BildNr].Spalten; s++)
            Picture[BildNr].Bild[z][s] =
                Grauwert[ Picture[BildNr].Bild[z][s] ];
}
```

5.5.2 Alias-Effekte

Die bei der Bildaufnahme erwähnten *Alias-Effekte* sind bei der Bildwiedergabe auf Rastergeräten ebenfalls zu beobachten. Der Begriff "Alias" stammt aus der Signaltheorie. Um eine Funktion fehlerfrei aus den abgetasteten Werten zurückgewinnen zu können, muß die Abtastfrequenz (*Nyquist-Frequenz*) mindestens doppelt so groß wie die höchste in der Funktion enthaltene Frequenz sein (Abtasttheorem von Shannon [Jer77]). Ansonsten überlappen sich die Spektren im Frequenzbereich (siehe Kapitel 8), das Bild wird unterabtastet und es treten Alias-Effekte auf. Die Abtastung eines hochfrequenten Signals kann dann die gleichen Werte wie die eines niederfrequenten Signals liefern. Man sagt dann, daß das niederfrequente Signal ein *Alias* des hochfrequenten Signals ist [Bli89b], [Bli89c]. Solche Phänomene können immer dann auftreten, wenn analoge Daten digitalisiert werden.

5. Geräte zur Bildwiedergabe

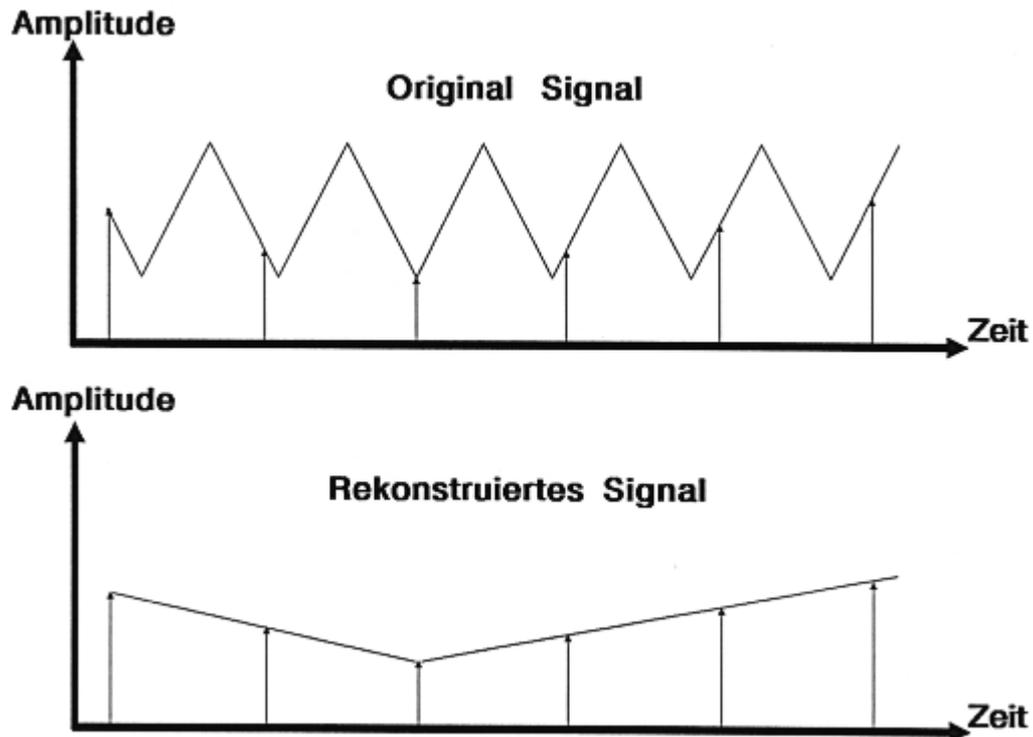


Abbildung 5.8: Wird ein Signal nicht mit genügend hoher Frequenz (Abtastrate) abgetastet, so können die Meßwerte dieselben sein wie bei der Abtastung eines niederfrequenten Signals. Ein "Alias" entsteht.

Eine Darstellung eines Objektes auf dem Rasterbildschirm, bei dem sich die Intensität entlang einer Linie plötzlich ändert, ist somit nicht fehlerfrei möglich. Diese plötzliche Änderung entspricht einer unendlich hohen Frequenz im Frequenzbereich (siehe Kapitel 8). Läßt sich die Bildschirmauflösung nicht erhöhen (entspricht der Erhöhung der Abtastfrequenz), so können durch Mittelung der Helligkeits-/Farbwerte über Nachbarbereiche bei Kanten diese störenden Effekte für den Betrachter etwas gemindert werden (*Anti-Aliasing*).

Je nach Art des Alias-Effektes unterscheidet man zwischen *Stairstepping* (stufenförmige Intensitätsänderungen entlang einer Linie, "Treppenstufen"), *Linebreakup* (Unterbrechungen entlang von Linien), *Crawling* (Erscheinen/Nicht-Erscheinen von dünnen Linien) und *Scintillation* (Größenveränderung bei der Bewegung von Objekten).

Möglichkeiten, diese störenden räumlichen Alias-Effekte zu beseitigen, bestehen in

- Erhöhung der Abtastrate,
- Präfilterung,
- Postfilterung.

Eine Erhöhung der Abtastrate ist in der Praxis aus technischen Gründen meist nicht möglich. Eine Präfilterung scheidet wegen des frühen Informationsverlustes (Filterung) und fehlender Kenntnis über die Geometrie der Szene aus. Daher wird in der Regel eine Postfilterung verwendet.

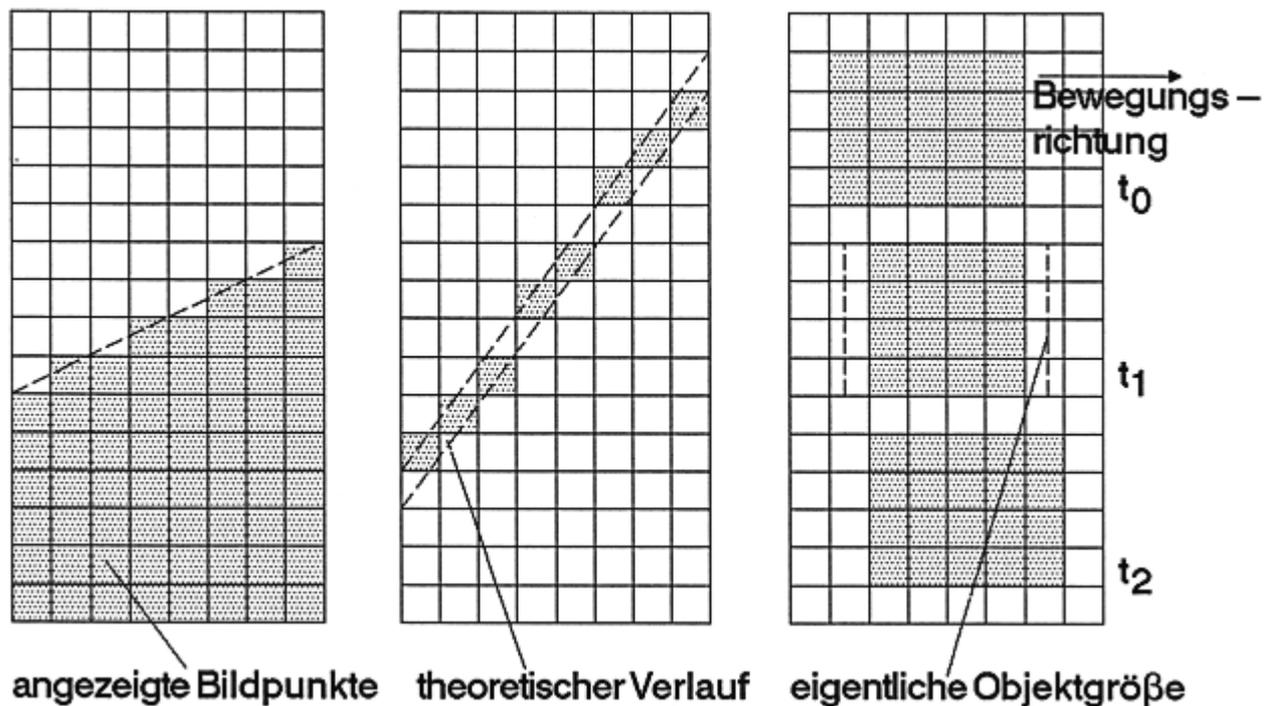


Abbildung 5.9: Beispiele für Alias-Effekte. Links: Bei schräg verlaufenden Linien entstehen aufgrund des diskreten Rasters Abstufungen (Stairstepping); Mitte: Wird ein Pixel nicht genügend durch eine Linie überdeckt, so können Lücken entstehen (Linebreakup); Rechts: Bei der Bewegung von Objekten kann durch die unterschiedliche Bedeckung der Pixel der Eindruck einer Größenveränderung entstehen (Scintillation). Durch entsprechende Filterungen können diese Effekte vermindert werden.

Eine Filterung bedeutet in dem Zusammenhang (vergleiche Kapitel 8 und Kapitel 10) eine Interpolation zwischen den diskreten Abtastwerten des Bildes. Die einfachste Lösung wäre dabei die Mittelung in einem Rechteckbereich (Box-Filter). Wie aber in Kapitel 8 deutlich wird, werden hierbei immer noch die hohen Frequenzen, teilweise etwas abgeschwächt, durchgelassen. Günstiger ist eine gewichtete Mittelung mit einem Filter in der Form einer Gaußkurve und ideal ein Filter in Form der *sinc*-Funktion:

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$$

Leider ist diese Filterfunktion unendlich weit. Eine räumliche Begrenzung führt wieder zu Fehlern und somit zu Alias-Effekten. Bei dieser Art von Filterung können auch negative Helligkeitswerte berechnet werden. Eine Begrenzung auf Null führt dann unter Umständen ebenfalls zu sichtbaren Fehlern. Bei Filterfunktionen, die über die Größe eines Bildpunktes hinausgehen, muß das Originalbild über die eigentliche Größe hinaus erweitert werden. Die außerhalb des Originalbildes verwendeten Werte führen in jedem Fall ebenfalls zu mehr oder minder deutlichen Fehlern [Bli89b].

5. Geräte zur Bildwiedergabe

Zur einfachen und schnellen Filterung wird daher in der Praxis mit einer theoretisch erhöhten Auflösung in Form von Subpixel-Bedeckungsmasken (meist 2×2 , ..., 8×8) gerechnet. Über diese Bedeckungsmasken kann die Fläche, die ein Objekt einen Bildpunkt überdeckt, approximiert werden [RW899]. Ein Subpixel wird als komplett bedeckt angesehen, falls der Subpixel-Mittelpunkt bedeckt ist. Der Intensitätswert des Bildpunktes berechnet sich dann aus der Berücksichtigung aller den Bildpunkt sichtbar überdeckenden Objektteile.

$$I'_{Pixel} = \sum_{i=1}^{Anzahl\ Objekte} A_i I_i + (1 - \sum_{i=1}^{Anzahl\ Objekte} A_i I_i) I_{Pixel}$$

wobei A_i die sichtbare Fläche ist, die das Objekt i den gerade betrachteten Bildpunkt bedeckt, I_i die Intensität bzw. Farbe des Objekts i und I_{Pixel} die Intensität des Bildpunktes (Hintergrund) sind. Der zweite Teil der Formel stellt somit die Fläche des Hintergrundes dar.

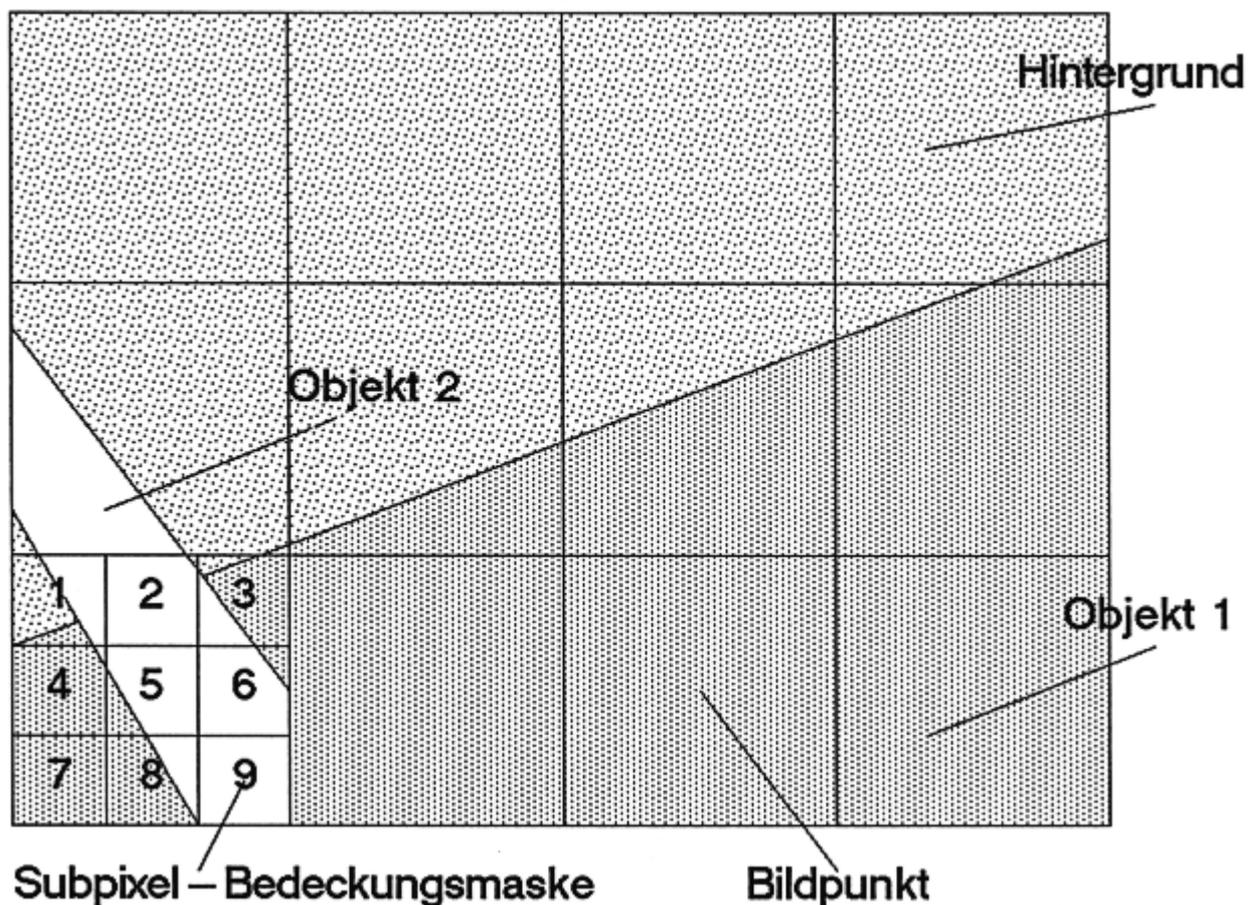


Abbildung 5.10: Die Gesamtintensität des Bildpunktes wird aus der Intensität der einzelnen Objekte, die diesen Bildpunkt teilweise überdecken, gewichtet mit der Überdeckungsfläche berechnet. Der Hintergrund zählt dabei auch dazu. Die einzelnen Flächen werden durch die Subpixel-Bedeckungsmaske approximiert.

Einen weiteren Ansatz zur Verminderung von Alias-Fehlern bieten auch die stochastischen Abtastverfahren, bei denen die Abtastpunkte nicht äquidistant, sondern stochastisch verteilt sind

5.5 Fehler bei der Bildwiedergabe und deren Beseitigung

[Coo86]. Der Vorteil, ein vermindertes Aliasing im diskreten Bild zu haben, wird hier durch Rauschen, bedingt durch die zufällige Abtastung, im abgetasteten Bild erkaufte, was der Betrachter aber als angenehmer empfindet, ebenso wie die Unschärfe, die durch die Filterung mit den Subpixel-Bedeckungsmasken entsteht.

Aufgaben

Aufgabe 1

Wie lange darf die Persistenz bei einem Monitor mit einer Bildwiederholfrequenz von 70 Hz maximal sein, damit das vorherige Bild nicht mehr erkennbar ist?

Aufgabe 2

Ein Schwarzweißmonitor hat bei einer Auflösung von 4096×3300 Pixel eine Bildwiederholfrequenz von 25 Hz. Wie groß ist die Bandbreite des Systems? Wie lang darf das Auslesen eines Bildpunktes aus dem Bildschirmspeicher maximal dauern?

Aufgabe 3

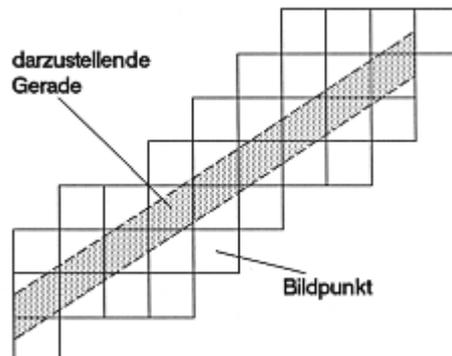
Ein 14"-Monitor (Seitenverhältnis 4:3) hat einen Lochmaskenabstand (Dot-Pitch) von 0.28 mm. Welche Auflösung kann damit maximal sinnvoll dargestellt werden?

Aufgabe 4

Berechnen Sie die Gamma-korrigierte Grauwerttabelle (LUT) für den Faktor $\gamma = 2.4$. Wieviel unterschiedliche Graustufen kann das System nach der Korrektur noch darstellen?

Aufgabe 5

Im folgenden Bild ist eine Gerade dargestellt (Breite 1 Pixel), die auf einem Rasterbildschirm ausgegeben werden soll.



Von welchen Faktoren hängt der jeweilige Grauwert eines Bildpunktes ab? Geben Sie allgemein an, wie sich der Grauwert der einzelnen Bildpunkte berechnet, wenn man Alias-Effekte möglichst vermeiden will.

6. Digitale Halbtonverfahren

Unter *digitalen Halbtonverfahren* werden Algorithmen verstanden, die Graustufen von Grauwertbildern durch Variation von Punktgröße oder Punktdichte in Binärbildern (nur die zwei Graustufen schwarz und weiß) umwandeln. Dies macht Sinn, da die vom menschlichen Auge noch wahrnehmbaren hell/dunkel-Wechsel bei starkem Kontrast ca. 100 Übergänge pro Sehwinkelgrad [CM87] betragen. Wird dieser Wert überschritten, so verschwimmen die Übergänge und der Eindruck von Grautönen entsteht. Diese Eigenart des menschlichen Sehsystems wird bei den Halbtonverfahren verwendet, wie sie in Zeitungen, Zeitschriften, Büchern usw. zur Wiedergabe von Grauton- und Farbbildern zu finden sind.

Bei einer Auflösung von 300 Punkten/Inch (üblich bei Laserdruckern) ergibt sich damit als geeigneter Betrachtungsabstand die Entfernung von mindestens 48.5 cm. Ist der Betrachtungsabstand fest, so läßt sich die maximal sinnvolle Auflösung berechnen. Eine Steigerung der Auflösung bringt in diesen Fällen keine subjektive Bildverbesserung.

Analog wird bei Farbbildern versucht, die Anzahl von Farbtönen ohne allzu großen Qualitätsverlust zu reduzieren. Durch geeignete Verfahren werden die am häufigsten im Bild vorkommenden Farben oder die zur Darstellung von Farbverläufen wichtigsten Farbtöne berechnet.

Insgesamt werden diese Verfahren bei der Ausgabe von Bildern auf Geräten mit niedrigerer Farb-/Grauwerttiefe angewandt, so zum Beispiel auf Monochrom-Monitoren, Druckern, aber auch zur Reduktion des Speicherbedarfs bei der Archivierung von Bildern. Alle hier im Buch abgedruckten Bilder sind mit solchen digitalen Halbtonverfahren aus Grauwertbildern erzeugt worden. Durch die Beschreibung der Halbtonverfahren wird die Aussagekraft der Abbildungen relativiert und einzelne Effekte werden besser verständlich.

6.1 Verfahren mit festem Schwellwert

Die einfachste Methode zur Überführung eines Grautonbildes in ein Halbtonbild (Binärbild) ist der bildpunktweise Vergleich mit einer konstanten Grauwertschwelle. Dieses einfache Verfahren kann man variieren, so daß als Ergebnis teilweise optisch schon recht ansprechende Bilder entstehen. Im folgenden wird o.B.d.A. davon ausgegangen, daß für den Grauwert I gilt:

$$0 \leq I \leq 255, \text{ mit } 0=\text{Schwarz und } 255=\text{Weiß}$$

dies gilt auch beim Binärbild.

6. Digitale Halbtonverfahren

6.1.1 Konstanter und angepaßter Schwellwert

Ist I_{max} der maximale im Bild vorkommende Grauwert, so gilt für die Grauwertschwelle T

$$T = I_{max} / 2$$

Wird in dem Bild nicht der gesamte Helligkeitsbereich von 0 bis 255 ausgenutzt, so ist es günstiger, die Schwelle T an die aktuell im Bild verwendeten Helligkeitswerte anzupassen

$$T = (\max_{x,y} \{I(x, y)\} + \min_{x,y} \{I(x, y)\}) / 2$$

Diese Vorgehensweise liefert dasselbe Ergebnis wie eine Spreizung der im Originalbild vorkommenden Grauwerte auf den ganzen zur Verfügung stehenden Grauwertbereich von 0 bis 255 (siehe Abschnitt 10.1). Hierbei wird wieder ausgenutzt, daß das menschliche Auge relative und nicht absolute Helligkeiten bewertet.

Nach Festlegung des Schwellwertes T berechnet sich das Binärbild $P(x, y)$ aus dem Grauwertbild $I(x, y)$ sehr einfach als

$$P(x, y) = \begin{cases} 255 & \text{falls } I(x, y) > T \\ 0 & \text{sonst} \end{cases}$$

Das Ergebnis dieser denkbar einfachsten Methode ist aber in den meisten Fällen sehr unbefriedigend, wie man an der Abbildung 6.1 erkennen kann. Viele Details gehen verloren und ein Eindruck von Graustufen ist nicht gegeben. Dieses einfache Verfahren eignet sich daher nur für eine Wiedergabe von Bildern, die ohnehin schon einen starken Kontrast bzw. binären Charakter aufweisen (Texte, Strichzeichnungen usw.).

```
void Schwellwert(int BildNr)
{
    int z,s;                /* Laufvariablen fuer Zeilen/Spalten */
    int Minimum, Maximum;  /* minimaler/maximaler Grauwert */
    int Schwelle;          /* ermittelte Schwelle */

    ClearTextWindow(35,18,80,25);
    WriteText(35,18,"* Schwellwertverfahren *");

    /* Zuerst minimaler und maximaler vorkommender Grauwert best. */
    Minimum=WEISS;
    Maximum=SCHWARZ;
    for (z=0; z<Picture[BildNr].Zeilen; z++)
        for (s=0; s<Picture[BildNr].Spalten; s++) {
            if (Picture[BildNr].Bild[z][s]<Minimum)
                Minimum=Picture[BildNr].Bild[z][s];
            else if (Picture[BildNr].Bild[z][s]>Maximum)
                Maximum=Picture[BildNr].Bild[z][s];
        }
}
```

```

/* Jetzt die angepasste Schwelle nach der Formel berechnen      */
Schwelle = (Maximum + Minimum) / 2;

/* die Berechnung des Binaerbildes erfolgt mit dieser Schwelle */
for (z=0; z<Picture[BildNr].Zeilen; z++)
  for (s=0; s<Picture[BildNr].Spalten; s++) {
    Picture[BildNr].Bild[z][s] =
      (Picture[BildNr].Bild[z][s]<=Schwelle) ? SCHWARZ : WEISS;
  }
}

```



Abbildung 6.1: Bei dem einfachen Schwellwertverfahren wird jeder Bildpunkt mit einer festen Schwelle verglichen. Liegt der Grauwert unter der Schwelle, erscheint der Bildpunkt schwarz, ansonsten weiß. Dieses Verfahren eignet sich nur bei Vorlagen mit starkem Kontrast.

6.1.2 Error-Diffusion

Durch das Setzen eines Bildpunktes auf schwarz oder weiß beim konstanten und festen Schwellwert wird pro Bildpunkt ein Fehler bezüglich des Grauwertes gemacht. Berücksichtigt man bei der Erzeugung eines Binärbildes diesen Fehler $E(x, y) = I(x, y) - P(x, y)$ und verteilt ihn auf die Nachbarpunkte, so kann man diesen Fehler kompensieren. Derartige Verfahren werden als *Error-Diffusion* bezeichnet. Zuerst wird dazu der erste Bildpunkt mit dem festen Schwellwert T verglichen. Danach wird der aufgetretene Fehler (Grauwertdifferenz) berechnet und auf die Nachbarpunkte verteilt. Diese Verteilung kann eine unterschiedliche Anzahl von Nachbarn betreffen [Stu81], [ES87].

Die Verteilung des Fehlers auf die m zu berücksichtigenden Nachbarn geschieht mit unterschiedlicher Gewichtung. Der neue Grauwert $I'_n(x, y)$ berechnet sich nach folgender Formel:

6. Digitale Halbtonverfahren

$$I'_n(x, y) = I_n(x, y) + \frac{\text{Gewicht von Nachbar } n}{\sum_{k=0}^{\text{alle zu berücksichtigende Nachbarn}} \text{Gewicht}(k)} E(x, y)$$

Für diese Formel und die nachfolgenden Ausführungen gilt:

n	n -ter Nachbarpunkt, auf den der Fehler verteilt wird
x	horizontaler Index (Spalte)
y	vertikaler Index (Zeile)
$I_n(x, y)$	Original-Grauert des n -ten Nachbarn
$I'_n(x, y)$	neuer Grauwert des n -ten Nachbarn
$E(x, y)$	an der Stelle (x, y) gemachter Fehler
X	bereits bearbeiteter Punkt
P	aktueller Punkt

Die Error-Diffusion liefert optisch recht ansprechende Ergebnisse, jedoch treten häufig Schatten ("Geisterbilder", "Echos") von im Originalbild enthaltenen Strukturen auf. Diese "Geisterbilder" entstehen dadurch, daß über viele Punkte hinweg ein Fehler weitergereicht wird, bis man schließlich den Schwellwert T überschreitet. Dieser störende Effekt kann gemindert werden, wenn die Summe der Gewichte kleiner als 1 ist (z.B. 0.9).

Aufwendiger aber wirkungsvoller ist eine von Floyd [Knu87] vorgeschlagene Reskalierung des Originalbildes. Der neue Grauwert $I'(i, j)$ berechnet sich dabei aus dem ursprünglichen Grauwert $I(i, j)$ nach der Formel

$$I'(i, j) = 25 + \text{INT}(0.8 I(i, j))$$

Wiederum wird dabei ausgenutzt, daß das menschliche Auge empfindlicher auf Kontraständerungen als auf Schwankungen im absoluten Grauwert reagiert.

Bei der Verteilung des gemachten Fehlers werden die horizontalen und vertikalen Nachbarn stärker berücksichtigt, um dort den Schwellwert schneller zu überschreiten und somit durchgehende orthogonale Strukturen zu vermeiden. Dadurch wird auch wieder eine der Eigenschaften des menschlichen Sehsystems berücksichtigt, horizontale und vertikale Strukturen zu bevorzugen, diagonale Verläufe dagegen mit einer um ca. 10% verminderten Intensität wahrzunehmen. Es erfolgt also keine gleichmäßige Verteilung des Fehlers auf die Nachbarn oder eine in Abhängigkeit des Abstandes zu dem gerade betrachteten Punkt.

Die Verteilung auf 4 Nachbarn erfolgt z.B. nach der vorgeschlagenen Gewichtung von Floyd und Steinberg [ES87]:

X	P	7	1/16	Error-Diffusion – 4 Nachbarn
3	5	1		

```

void Error_Diff(int BildNr)
{
    int z,s;                /* Laufvariablen fuer Zeilen/Spalten */
    int Grauwert;          /* Hilfsvariable fuer die Error-Diff. */
    float Error;           /* gemachter Fehler bei der Error-D. */
    int Schwelle;          /* Verwendete Schwelle */

    ClearTextWindow(35,18,80,25);
    WriteText(35,18,"* Error-Diffusion *");

    /* Initalisierung der Schwelle. Ggf. angepasster Schwellwert */
    Schwelle=GRAU;

    /* Error-Diffusion ueber das ganze Bild durchfuehren */
    for (z=0; z<Picture[BildNr].Zeilen; z++)
        for (s=0; s<Picture[BildNr].Spalten; s++) {
            Grauwert =
                (Picture[BildNr].Bild[z][s]<=Schwelle) ? SCHWARZ:WEISS;
            /* Fehler berechnen */
            Error = Picture[BildNr].Bild[z][s] - Grauwert;
            /* aktuellen Bildpunkt auf den Binaer-Wert setzen */
            Picture[BildNr].Bild[z][s]=Grauwert;

            /* Fehler jetzt gewichtet auf die 4 Nachbarn verteilen */
            if ((s+1) < Picture[BildNr].Spalten) { /* Rand beachten */
                Picture[BildNr].Bild[z][s+1] += Error * 7.0/16.0;
                if ((z+1) < Picture[BildNr].Zeilen) /* Rand beachten */
                    Picture[BildNr].Bild[z+1][s+1] += Error * 1.0/16.0;
            }
            if ((z+1) < Picture[BildNr].Zeilen) {
                Picture[BildNr].Bild[z+1][s] += Error * 5.0/16.0;
                if (0 < s)
                    Picture[BildNr].Bild[z+1][s-1] += Error * 3.0/16.0;
            }
            /* Anmerkung: Falls die Bilder vom Typ "unsigned char" */
            /* sind, kann die Berechnung so nicht erfol- */
            /* gen. Bei der Fehlerverteilung entstehen */
            /* Werte ausserhalb des Wertebereiches. */
            /* Die Fehlerverteilung muss dann in einem */
            /* zweizeiligen Hilfsarray vom Typ "int" */
            /* durchgefuehrt werden! */
        }
    }
}

```

Zur leichteren Berechnung wurden die unterschiedlichen Gewichte als Zweierpotenzen realisiert [Stu81]. Man erhält dann folgende Verteilung auf 6 Nachbarn:

X	P	8	2	1/24	Error-Diffusion – 6 Nachbarn
2	8	2			
	2				

6. Digitale Halbtonverfahren



Abbildung 6.2: Error-Diffusion mit 4 Nachbarn (links) und mit 12 Nachbarn (rechts). Bei der Verteilung des Fehlers auf noch mehr Nachbarpunkte treten weniger störende Texturen bei Gebieten mit kontinuierlicher Grauwertänderung ("Graukeil") oder konstantem Grauwert ("Himmel") auf.

Werden 12 Nachbarn berücksichtigt, auf die der Fehler verteilt wird, erhöht sich zwar der Rechenaufwand, dafür verschwinden die störenden Texturen fast vollständig, wie Abbildung 6.2 zeigt. Die Verteilung erfolgt nach:

X	X	P	8	4
2	4	8	4	2
1	2	4	2	1

1/42 Error-Diffusion – 12 Nachbarn

Eine Verteilung auf noch mehr Nachbarn bringt optisch keine Verbesserung mehr [Stu81].

6.2 Verfahren mit variablem Schwellwert

Bei Verfahren mit variablem Schwellwert T (*Dither-Verfahren*) wird T von Bildpunkt zu Bildpunkt geändert. Es existieren hier verschiedene Varianten, die sich darin unterscheiden, wie der Schwellwert pro Bildpunkt berechnet wird.

Eine denkbare Möglichkeit besteht darin, den Schwellwert T pro Bildpunkt mit einem Zufallszahlengenerator zu erzeugen. Solche Bilder beinhalten aber sehr viel Rauschen. Die Detailauflösung ist ebenfalls gering. Aus diesem Grunde werden feste Matrizen mit Schwellwerten verwendet. Durch den speziellen Aufbau dieser Matrizen läßt sich die räumliche Auflösung u.ä. beeinflussen.

6.2.1 Ordered-Dither

Beim *Ordered-Dither* handelt es sich bei T nicht mehr um einen einzelnen Zahlenwert, sondern um eine Matrix $T(i, j)$. Der Schwellwert für jeden Bildpunkt wird dann durch den Wert an der korrespondierenden Stelle in der Matrix $T(i, j)$ bestimmt [ES87].

Der Grauwert $I(x, y)$ wird mit dem Schwellwert $T(i, j)$ verglichen, wobei

$$i = x \bmod N$$

$$j = y \bmod N$$

ist und die Dithermatrix $T(i, j)$ die Größe $N \times N$ besitzt. Die in dieser Dithermatrix vorkommenden Schwellwerte sind die Zahlen von 0 bis N^2-1 . Diese Werte müssen dann noch an die aktuelle Grauwertskala angepaßt werden (z.B. bei einer 8×8 -Matrix mit 64 Werten und 256 Graustufen müssen die Einträge mit 4 multipliziert werden).

In den Dithermatrizen müssen die Elemente so angeordnet werden, daß störende Texturen weitgehend vermieden werden. Ein gutes Schema zur rekursiven Erzeugung solcher Matrizen wurde von Bayer vorgeschlagen [ES87].

$$T_{2N}(i, j) = \begin{bmatrix} 4T_N & 4T_N + 2U_N \\ 4T_N + 3U_N & 4T_N + U_N \end{bmatrix}$$

Mit dieser Formel lassen sich Matrizen der Größe 2, 4, 8, ... erzeugen. Dabei ist U_N eine $N \times N$ -Matrix, deren Elemente alle 1 sind. Für $N=2$ hat die resultierende Matrix folgende Gestalt:

$$T_4(i, j) = \begin{bmatrix} 0 & 8 & 2 & 10 \\ 12 & 4 & 14 & 6 \\ 3 & 11 & 1 & 9 \\ 15 & 7 & 13 & 5 \end{bmatrix}$$

Eine 8×8 -Dithermatrix mit optimiertem Aufbau bezüglich der räumlichen Auflösung (abwechselnd große und kleine Zahlenwerte in der Matrix) ist in [Stu81] vorgestellt worden und hat folgende Gestalt:

$$T(i, j) = \begin{bmatrix} 22 & 6 & 18 & 2 & 21 & 5 & 17 & 1 \\ 14 & 30 & 10 & 26 & 13 & 29 & 9 & 25 \\ 20 & 4 & 24 & 8 & 19 & 3 & 23 & 7 \\ 12 & 28 & 16 & 32 & 11 & 27 & 15 & 31 \\ 21 & 5 & 17 & 1 & 22 & 6 & 18 & 2 \\ 13 & 29 & 9 & 25 & 14 & 30 & 10 & 26 \\ 19 & 3 & 23 & 7 & 20 & 4 & 24 & 8 \\ 11 & 27 & 15 & 31 & 12 & 28 & 16 & 32 \end{bmatrix}$$

6. Digitale Halbtonverfahren

Um Details besser hervorzuheben, empfiehlt sich in vielen Fällen eine Kantenverstärkung (siehe Abschnitt 10.8). Der Kontrast wird bei möglichen Kanten erhöht, was eine Erhöhung der Detailauflösung durch das Auge mit sich bringt [CM87]. Eine zu starke Betonung von Kanten kann aber zu störendem Rauschen im Bild führen. Für die Kantenverstärkung kann prinzipiell jeder Kantenfiter (siehe Abschnitt 10.8 und Kapitel 11) eingesetzt werden.

Bedingt durch die Punktform und die Anforderung, auch komplett schwarze Flächen beim Ausdruck erzeugen zu müssen, überlappen sich bei der Ausgabe auf einem Drucker die einzelnen Punkte leider etwas. Diese Überlappung führt dazu, daß der Schwärzungsgrad nicht konstant mit zunehmender Helligkeit abnimmt. Die auf solchen Geräten ausgegebenen Bilder wirken daher dunkler.

Die obige 8×8-Dithermatrix berücksichtigt die Überlappung von nebeneinander liegenden Punkten nicht. Beachtet man diese Überlappungen und versucht, die Reihenfolge der Schwellen in kleinen Gruppen anzuordnen, so hat die diesbezüglich optimierte Matrix folgende Gestalt [Stu81]:

$$T(i, j) = \begin{bmatrix} 19 & 25 & 23 & 17 & 14 & 8 & 10 & 16 \\ 21 & 31 & 29 & 27 & 12 & 2 & 4 & 6 \\ 28 & 30 & 32 & 22 & 5 & 3 & 1 & 11 \\ 18 & 24 & 26 & 20 & 15 & 9 & 7 & 13 \\ 14 & 8 & 10 & 16 & 19 & 25 & 23 & 17 \\ 12 & 2 & 4 & 6 & 21 & 31 & 29 & 27 \\ 5 & 3 & 1 & 11 & 28 & 30 & 32 & 22 \\ 15 & 9 & 7 & 13 & 18 & 24 & 26 & 20 \end{bmatrix}$$

Die Reihenfolge der einzelnen Schwellen und die der Gruppen ist auch hier wieder in einer diagonalen Struktur angeordnet, um den Eigenschaften des menschlichen Sehsystems Rechnung zu tragen.

```
void Ordered_Dither(int BildNr)
{
    int z,s;                /* Laufvariablen fuer Zeilen/Spalten */
    int Matrix_Groesse;    /* Groesse der Dither-Matrix */
    int Klasse;            /* Hilfsvariablen fuer die Klassen */
    /* 8x8-Matrix - Optimiert bezueglich der Aufloesung. Diese */
    /* Matrix ist eine Alternative zu der im Text vorgestellten. */
    unsigned char Muster[8][8]={ { 0, 32, 8, 40, 2, 43, 10, 42},
                                  { 48, 16, 56, 24, 50, 18, 58, 26},
                                  { 12, 44, 4, 36, 14, 46, 6, 38},
                                  { 60, 28, 52, 20, 62, 30, 54, 22},
                                  { 3, 35, 11, 43, 1, 33, 9, 41},
                                  { 51, 19, 59, 27, 49, 17, 57, 25},
                                  { 15, 47, 7, 39, 13, 45, 5, 37},
                                  { 63, 31, 55, 23, 61, 29, 53, 21} };
}
```

```

ClearTextWindow(35,18,80,25);
WriteText(35,18,"* Ordered-Dither *");

/* Initialisierung der Variablen. Anpassung an Grauwertbereich */
Matrix_Groesse = 8;
for (z=0; z<Matrix_Groesse; z++)
    for (s=0; s<Matrix_Groesse; s++)
        Muster[z][s] *= 4;

/* Den Ordered-Dither-Vorgang mit Vergleich der Klassen starten*/
for (z=0; z<Picture[BildNr].Zeilen; z++)
    for (s=0; s<Picture[BildNr].Spalten; s++) {
        /* Berechnung der Klasse durch Modulo-Rechnung */
        Klasse = Muster[z % Matrix_Groesse][s % Matrix_Groesse];
        Picture[BildNr].Bild[z][s] =
            (Picture[BildNr].Bild[z][s]>Klasse) ? WEISS: SCHWARZ;
    }
}

```



Abbildung 6.3: Beispiel für das Ergebnis des Ordered-Dither Verfahrens mit der grauwertoptimierten zweiten 8×8-Dithermatrix. Deutlich ist der Verlust an Detailauflösung zu erkennen.

6.2.2 Lokaler Schwellwert

In [JR76] wurde ein Verfahren vorgestellt, das den aktuellen Schwellwert für jeden Bildpunkt in Abhängigkeit der Grauwerte der 8 Nachbarn berechnet.

$$T = \gamma + \overline{I(i, j)} * \left(1 - \frac{2\gamma}{I_{\max}}\right)$$

6. Digitale Halbtonverfahren

wobei gilt

T	= Schwellwert
γ	= Konstante
$\overline{I(i, j)}$	= Mittlerer Grauwert im 3×3 -Feld
I_{max}	= Maximaler im Bild vorkommender Grauwert

Der Faktor γ bestimmt das Rauschen und den Kontrast. Bei $\gamma = 0$ wirkt das Ergebnis stark verrauscht. Für $\gamma = I_{max}/2$ erhält man das normale Schwellwertverfahren. Versuche haben gezeigt, daß Werte zwischen 4 und 6 für γ die besten Ergebnisse liefern. Kleinere Werte verrauschen das Bild stärker, größere liefern zwar ein kontrastreicheres Bild, Details gehen aber verloren.

6.3 Grautonmuster

Werden Ausgabegeräte verwendet, bei denen die Punktauflösung wesentlich besser ist als die des menschlichen Auges, können die Graustufen durch unterschiedliche Punktgrößen realisiert werden. Unter anderem beim Zeitungsdruck wird dieses Verfahren verwendet. Die Auflösung beträgt dort etwa 60-80 Punkte/Inch. Für Zeitschriften und Bücher werden bis zu 150 Punkte/Inch verwendet, bei Fotosatzqualität erreicht man durchaus Auflösungen bis zu 2400 Punkte/Inch.

Bei den üblicherweise zur Verfügung stehenden Druckern und Sichtgeräten ist die Punktgröße innerhalb einer gewissen Toleranz konstant [Stu81]. Man muß dort daher die Größenvariationen durch eine unterschiedliche Anzahl und Dichte von Punkten simulieren [FvD84]. Ein Feld von $N \times N$ Punkten bildet dann den neuen, in der Größe veränderbaren "Bildpunkt". Abbildung 6.4 und 6.5 zeigen solche *Grautonmuster* mit 5 bzw. 10 Grauwerten.

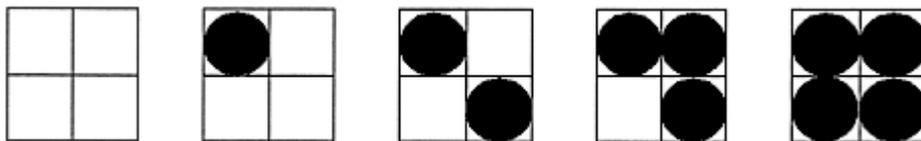


Abbildung 6.4: Zur Simulation von 5 Graustufen ist eine 2×2 -Matrix je Bildpunkt notwendig. Zur Vermeidung von Artefakten werden die diagonalen Strukturen auch hier bevorzugt.

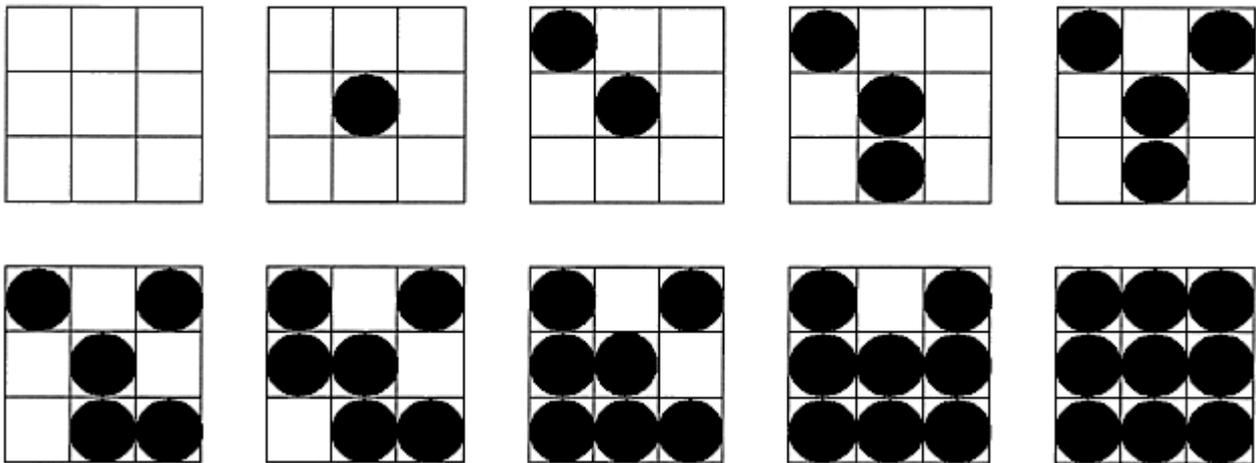


Abbildung 6.5: Die spezielle Anordnung der einzelnen Punkte verhindert bei diesem Grautonmuster mit 10 Graustufen ebenfalls unerwünschte Strukturen im Ausgabebild.

Denkbar sind noch andere Verteilungen der einzelnen Punkte in einem solchen $N \times N$ -Feld, jedoch sollten die Punkte immer ein zusammenhängendes Gebiet darstellen, um die Punktgrößenmodulation zu simulieren und dabei keine Richtung besonders zu bevorzugen. Mit einem $N \times N$ -Feld können $N^2 + 1$ Graustufen repräsentiert werden. Durch die regelmäßigen Muster entstehen aber im Bild unerwünschte Strukturen. Diese kann man zu Lasten der Ortsauflösung durch Grautonmuster mit mehr Graustufen etwas mindern. Es bleibt dabei immer der Zwiespalt zwischen wachsender Anzahl darstellbarer Graustufen und Verlust an räumlicher Auflösung. Die Qualität der variablen Punktgrößenmodulation wird mit diesen Methoden nicht erreicht.

```
void Grauton_Muster(int BildNr)
{
    int z,s;                /* Laufvariablen fuer Zeilen/Spalten */
    int i,j;                /* Indexvariablen fuer Grauton-Muster */
    unsigned char Grauwert; /* Pixel-Grauwert */
    int GrautonMuster;     /* zur Graustufe zugehoeriges Muster */
    unsigned char OutputPic[3*ZEILEN][3*SPALTEN]; /* Ausgabebild */
    /* 3x3-Matrix fuer die Grauton-Muster der 10 Graustufen. */
    /* Matrix ist eine Alternative zu der im Text vorgestellten. */
    unsigned char Muster[10][3][3]={ { {SCHWARZ,SCHWARZ,SCHWARZ},
                                        {SCHWARZ,SCHWARZ,SCHWARZ},
                                        {SCHWARZ,SCHWARZ,SCHWARZ}}, },
                                      { {SCHWARZ,SCHWARZ,SCHWARZ},
                                        {SCHWARZ,WEISS ,SCHWARZ},
                                        {SCHWARZ,SCHWARZ,SCHWARZ}}, },
                                      ...
                                      { {WEISS ,SCHWARZ,WEISS },
                                        {WEISS ,WEISS ,WEISS },
                                        {WEISS ,WEISS ,WEISS }}, },
                                      { {WEISS ,WEISS ,WEISS },
                                        {WEISS ,WEISS ,WEISS },
                                        {WEISS ,WEISS ,WEISS } } }
```

6. Digitale Halbtonverfahren

```
ClearTextWindow(35,18,80,25);
WriteText(35,18,"* Grauton-Muster *");

for (z=0; z<Picture[BildNr].Zeilen; z++)
  for (s=0; s<Picture[BildNr].Spalten; s++) {
    Grauwert = Picture[BildNr].Bild[z][s];
    /* Index des zugehoerigen Musters ermitteln */
    GrautonMuster = (int)( (float)Grauwert / 25.5 );

    /* Nun Muster in 3x so grosses Ausgabebild uebertragen */
    for (i=0; i<3; i++)
      for (j=0; j<3; j++)
        OutputPic[3*z+i][3*s+j] =
          Muster[GrautonMuster][i][j];
  }
/* Ergebnisbild nun nach Belieben anzeigen, ausdrucken usw. ...*/
}
```



Abbildung 6.6: Dieses Binärbild wurde mit einem Grautonmuster mit 10 Stufen erzeugt. Die Abstufung ist deutlich im Graukeil zu erkennen.

6.4 Kombinierte Verfahren

Neben der Möglichkeit, die einzelnen Halbtonverfahren mit den unterschiedlichen Vorverarbeitungsschritten zu kombinieren, ist es auch möglich, die Halbtonalgorithmen an sich zu kombinieren. Es ist z.B. denkbar eine Error-Diffusion nicht mit zwei Stufen (schwarz oder weiß), sondern mit fünf Stufen anzuwenden. Diese fünf Graustufen kann man dann mit Grautonmuster in einem 2×2 -Feld repräsentieren.



Abbildung 6.7: Dieses Bild wurde mit Error-Diffusion auf 4 Nachbarn unter Berücksichtigung von 10 Graustufen erzeugt. Die 10 Graustufen sind danach über Grautonmuster erzeugt und gedruckt worden.

6.5 Dot-Diffusion

Das Verfahren des Ordered-Dither läßt sich leicht implementieren und erzeugt auch keine Schattenbilder, jedoch werden die Bilder vergrößert. Strukturen, die kleiner als N sind (bei einer $N \times N$ -Dithermatrix) können ganz verschwinden. Die Detailauflösung nimmt ab. Error-Diffusion liefert zwar optisch relativ gute Bilder, hat aber dafür den Nachteil der "Geisterbilder". Man hat daher versucht, die Vorteile dieser beiden Verfahren in der *Dot-Diffusion* zusammenzufassen [Knu87].

6. Digitale Halbtonverfahren

Bei der Dot-Diffusion wird -- wie beim Ordered-Dither -- eine $N \times N$ Matrix $C(i, j)$ mit Zahlen von 0 bis N^2-1 verwendet (üblicherweise $N = 8$). Über die (x, y) -Koordinaten eines jeden Bildpunktes werden wieder gemäß

$$i = x \bmod N$$

$$j = y \bmod N$$

Indizes für die Matrix $C(i, j)$ berechnet. Jeder Bildpunkt erhält somit zu seinem Grauwert über diese Matrix noch einen neuen Zahlwert, der seine Klasse repräsentiert. Dadurch werden die Punkte des Originalbildes in N^2 Klassen eingeteilt. Das Bild wird jetzt nicht mehr – wie bei den bisherigen Verfahren zeilenweise, von links nach rechts – abgearbeitet, sondern die Reihenfolge wird durch die Klassenzugehörigkeit, beginnend mit Klasse 0, festgelegt. Die Reihenfolge, in der die Punkte derselben Klasse behandelt werden, spielt keine Rolle. Analog zur Error-Diffusion wird jetzt anhand eines konstanten Schwellwertes T entschieden, ob ein Punkt auf schwarz oder auf weiß gesetzt werden soll. Der hierbei auftretende Fehler wird nun auf jene Nachbarn des Punktes verteilt, die einer höheren Klasse angehören, also noch zu bearbeiten sind. Es ist wiederum vorteilhaft, die orthogonalen Nachbarn gegenüber den diagonalen zu bevorzugen.

Die Anforderungen an die Matrix bei der Dot-Diffusion unterscheiden sich grundsätzlich von denen beim Ordered-Dither. Die Zahlen sind in der Matrix so zu verteilen, daß möglichst wenig Matrixelemente ausschließlich Nachbarn haben, die einer niedrigeren Klasse angehören. Der Fehler solcher Matrixelemente kann in diesen Fällen nicht mehr weitergegeben werden. Den Aufbau einer Matrix, die nur ein derartiges Matrixelement, das letzte der Klasse enthält, lautet wie folgt [Ku87]:

$$C(i, j) = \begin{bmatrix} 25 & 21 & 13 & 39 & 47 & 57 & 53 & 45 \\ 48 & 32 & 29 & 43 & 55 & 63 & 61 & 56 \\ 40 & 30 & 35 & 51 & 59 & 62 & 60 & 52 \\ 36 & 14 & 22 & 26 & 46 & 54 & 58 & 44 \\ 16 & 6 & 10 & 18 & 38 & 42 & 50 & 24 \\ 8 & 0 & 2 & 7 & 15 & 31 & 34 & 20 \\ 4 & 1 & 3 & 11 & 23 & 33 & 28 & 12 \\ 17 & 9 & 5 & 19 & 27 & 49 & 41 & 37 \end{bmatrix}$$

6.6 Digitale Halbtonverfahren bei Farbbildern

Ebenso, wie Grauwertbilder in solche mit weniger Graustufen umgerechnet werden können, kann auch die Anzahl der Farben in Farbbildern zur Ausgabe oder Archivierung reduziert werden. Dies kann z.B. notwendig sein, um Echtfarbenbilder (8-Bit je Farbe; insgesamt $(2^8)^3 = 2^{24}$, also über 16 Millionen Farben) auf Geräten mit nur 256 möglichen Farben auszugeben.

Weiterhin besteht die Möglichkeit, die in den Bildern verwendeten Farben in Helligkeitswerte umzurechnen und die so erzeugten Grauwertbilder mit den vorher beschriebenen digitalen Halbtonverfahren in Binärbilder umzurechnen.

6.6.1 Ausgabe von Farbbildern als Binärbilder

Das menschliche Auge ist für die einzelnen Farben unterschiedlich empfindlich [FvD84], [ES87], d.h., bei energiegleicher Strahlung hat der Mensch bei unterschiedlichen Wellenlängen des Lichtes ein unterschiedliches Helligkeitsempfinden. Man kann somit aus den Farben die Luminanz Y (Helligkeit) berechnen, was beim Farb- bzw. Schwarzweiß-Fernsehen ausgenutzt wird. Die Luminanz enthält dann nur die Schwarzweißinformation bzw. Helligkeit und berechnet sich pro Pixel als gewichtete Summe von:

$$Y = 0.3R + 0.59G + 0.11B$$

mit Y =Luminanz, R =Rot-Anteil, G =Grün-Anteil und B =Blau-Anteil. Aus dieser Umrechnung resultiert ein Grauwertbild, das in etwa die subjektive Helligkeit des Farbbildes wiedergibt. Auf dieses können dann die schon bekannten Algorithmen angewandt werden.

Eine andere Möglichkeit, RGB-Werte in Helligkeitswerte umzurechnen, bietet das HLS-System (H=Farbton (hue), L=Helligkeit (lightness) und S=Sättigung (saturation)) [FvD85], [ES87]. Hier wird die Helligkeit nach der Formel

$$L = (\max(R, G, B) + \min(R, G, B)) / 2$$

berechnet. Da die einzelnen Farben als gleichmäßig gewichtet betrachtet werden, bleibt der subjektive Helligkeitseindruck unberücksichtigt und die Bilder wirken dunkler.

6.6.2 Farbdithering

Für die Reduktion der Farbanzahl müssen die wichtigsten Farben in einem Bild berechnet werden. Was unter "wichtigsten" zu verstehen ist, hängt von dem Einsatzgebiet und dem gewählten Algorithmus ab.

Stehen statt den 3×256 Farbintensitäten nur 256-Farbtabelle-Einträge zur Verfügung, so können für eine einfache Umrechnung nur jeweils 6 unterschiedliche Helligkeiten je Farbe verwendet werden. Insgesamt ergeben sich $6 \times 6 \times 6 = 216$ Farben mit 6 Graustufen. Bei der linearen Umrechnung können jedoch Farben entstehen, die im Originalbild nicht vorkommen. Kontinuierliche Farbübergänge bekommen sichtbare Abstufungen.

Besser ist die Verwendung einer Häufigkeitstabelle und der Aufbau der Farbtabelle mit den 256 häufigsten im Bild vorkommenden Farben. Farben für wichtige Details können aber dabei verloren gehen.

6. Digitale Halbtonverfahren

Eine weitere Verbesserung wird erreicht, wenn jeder Eintrag in der Farbtabelle in etwa die gleiche Anzahl von Bildpunkten im Originalbild repräsentiert. Dies erfolgt mit dem sogenannten *Median-Cut-Verfahren* [Hec82]. Der RGB-Würfel (mit den Achsen Rot, Grün und Blau) wird dabei nicht gleichmäßig aufgeteilt, sondern es wird zuerst der kleinste Quader im RGB-Raum bestimmt, der alle Farben des Originalbildes enthält. Dieser Quader wird an der größten Seite am Median in zwei neue Quader aufgesplittet. Diese enthalten in etwa gleich viele Bildpunkte. Danach werden diese Quader wieder aufgeteilt. Diese Aufteilung der Quader mit den meisten Bildpunkten am Median erfolgt solange, bis die Anzahl von 256 Quadern erreicht ist. Von jeder der 256 erzeugten Boxen wird der Schwerpunkt ermittelt, der dann die Farbe für diesen Quader in der Farbtabelle darstellt. Wie diese Beschreibung schon erkennen läßt, erfordert das Median-Cut-Verfahren einen großen Rechenaufwand. Das Ergebnis des *Farbditherings* ist aber auch das beste der hier vorgestellten Verfahren.

Aufgaben

Aufgabe 1

Gegeben sei folgender Ausschnitt eines Grauwertbildes (Grauwert als Zahlen von 0...255 wiedergegeben):

250	0	7	230	10	8	200	30	194
200	5	1	189	13	9	170	28	169
170	9	8	180	23	19	145	48	123
128	128	128	128	128	128	128	128	128

Bildausschnitt

- Überführen Sie dieses Grauwertbild in ein Binärbild durch Verwendung eines einfachen Schwellwertverfahrens mit der Schwelle $T = 127$. Wie sieht der Ausschnitt danach aus?
- Welches Ergebnis erhält man, wenn das Error-Diffusion-Verfahren mit der Verteilung des Fehlers auf 4 Nachbarn ($T = 127$) auf den Ausschnitt angewandt wird?

X	P	7
3	5	1

Dithermatrix

- Welche Vorkehrungen sind am Rand zu berücksichtigen?
- Begründen Sie den Aufbau der Dithermatrix.

Aufgabe 2

Entwickeln Sie alternative Grautonmuster zur Simulation von 10 Graustufen. Beachten Sie dabei die Hinweise zur Vermeidung von unerwünschten Strukturen.

6. Digitale Halbtonverfahren

7. Bildrestaurierung

Unter *Bildrestaurierung* versteht man die Verbesserung von Bildsignalen im Sinne quantitativ definierter Kriterien (im Gegensatz zu den subjektiven Kriterien, die zum Teil schon angesprochen wurden und noch im Kapitel 10 angesprochen werden.).

Bei der Bildrestaurierung hat man a-priori-Wissen über die Art der Störungen. Quellen für solche Störungen sind z.B. die schon besprochenen Fehler bei der Aufnahme und Wiedergabe, aber auch Störungen durch andere, bekannte Signale. Diese Störungen werden, sofern möglich, separat gemessen. Dann versucht man, diese mathematisch zu beschreiben. Durch Anwendung der inversen Störfunktion wird danach angestrebt, das Originalbild ohne Fehler zurückzuerhalten. Lassen sich die Störungen nicht so einfach messen, so wird eine Funktion bestimmt, die den Fehler minimiert (im Sinne der kleinsten Fehlerquadrate). Diese Funktion wird dann zur Bildrestaurierung verwendet.

7.1 Beispiele für Störungen

Neben den schon erwähnten Fehlern bei der Bildaufnahme bzw. Bildwiedergabe können noch zusätzlich Störungen vorkommen, z.B. in Form von Verschmierung aufgrund von Bewegungen der Aufnahmegerate bzw. der Objekte oder durch additive Signale, die Helligkeits- und Kontrastwerte ändern. Diese Störungen lassen sich in der Regel durch lineare, positionsinvariante Prozesse approximieren.

Anders ist es bei periodischen Störungen, wie z.B. Brummen, jedes n -te Pixel ist durch einen Fehler im CCD-Chip gestört, oder bei Störungen, wie sich örtlich verändernde Auflösungen, und Überlagerungen des Bildsignals mit signalabhängigen Störungen.

7.2 Signalmodell zur Bilddegradation

Der Störungsprozeß sei im folgenden durch den Operator H dargestellt, der zusammen mit einem additiven, unkorrelierten Rauschterm $\eta(x, y)$ auf dem ungestörten Eingangssignal $f(x, y)$ arbeitet und das gestörte (degradierte) Ausgangssignal $g(x, y)$ liefert.

$$g(x, y) = H f(x, y) + \eta(x, y)$$

Ist der Operator H positionsinvariant, so ist das Ergebnis an einem Punkt im Bild nur vom Eingangswert an dieser Stelle und nicht von der Position abhängig. Zur Vereinfachung wird oft $\eta(x, y) = 0$ gesetzt.

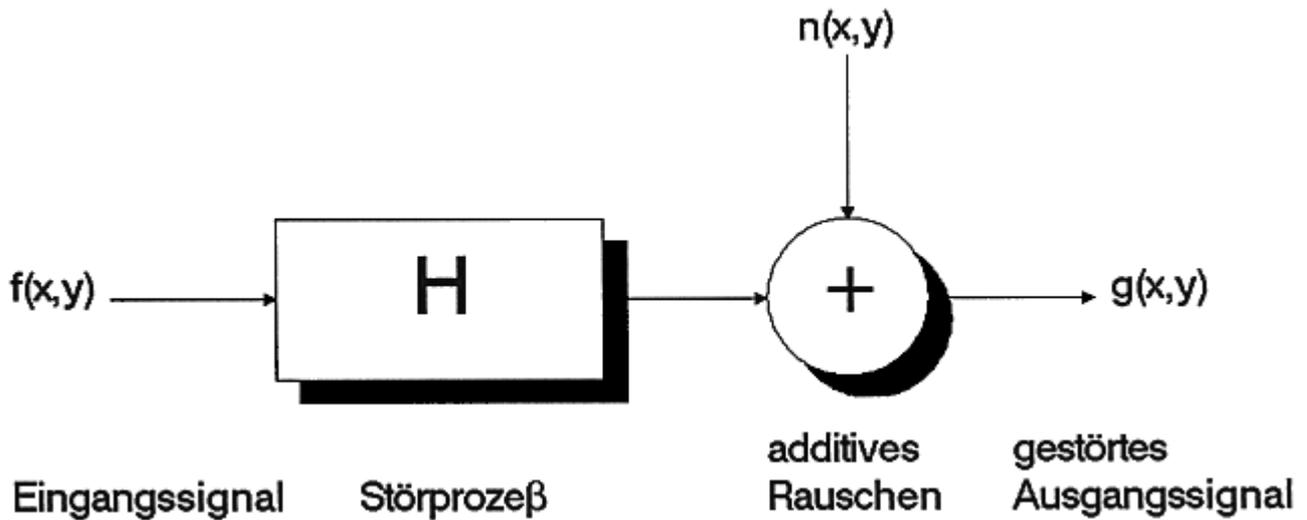


Abbildung 7.1: Ein Eingangssignal $f(x, y)$ wird durch einen positionsinvarianten Störprozeß H verändert und additiv mit Rauschen $\eta(x, y)$ versehen. Daraus resultiert das gestörte Signal $g(x, y)$. Mittels einer entsprechenden inversen Filterung soll dann ein möglichst "optimales" Eingangssignal \hat{f} restauriert werden.

7.2.1 Kontinuierliche Funktion

Die Abtastung eines zweidimensionalen kontinuierlichen Signals soll mit Hilfe der zweidimensionalen Delta-Funktion $\delta(x, y)$ geschehen. Diese Delta-Funktion ist wie folgt definiert (vergleiche Kapitel 8):

$$\delta(x, y) = \begin{cases} 1 & \text{für } x, y=0 \\ 0 & \text{sonst} \end{cases}$$

Der Abstand der diskreten Abtaststellen wird durch α und β beschrieben.

$$\delta(x - \alpha, y - \beta) = \begin{cases} 1 & \text{für } x = \alpha, y = \beta \\ 0 & \text{sonst} \end{cases}$$

Für die Abtastung der kontinuierlichen Funktion ergibt sich damit

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) \delta(x - \alpha, y - \beta) d\alpha d\beta$$

und somit für das gestörte Ausgangssignal

$$g(x, y) = H f(x, y) + \eta(x, y) = H \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) \delta(x - \alpha, y - \beta) d\alpha d\beta + \eta(x, y)$$

Da H ein linearer Operator ist, also die Antwort auf die Summe zweier Eingangssignale gleich der Summe der zwei Antworten (*additive Eigenschaft*) sowie die Antwort auf ein mit einer Konstanten

multipliziertes Eingangssignal gleich der Antwort des Eingangssignals multipliziert mit der Konstanten (*Homogenität*) ist, gilt

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) H \delta(x - \alpha, y - \beta) d\alpha d\beta + \eta(x, y)$$

Der Ausdruck $H\delta(x - \alpha, y - \beta) = h(x, \alpha, y, \beta)$ wird auch als *Impuls-Antwort* von H bezeichnet. Ist H von der Position unabhängig, gilt

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) h(x - \alpha, y - \beta) d\alpha d\beta + \eta(x, y)$$

(Dieser Ausdruck entspricht dem Faltungsintegral bei der Fourier-Transformation). Ist die Impuls-Antwort bekannt, so kann die Antwort (gestörtes Signal) auf jedes beliebige Eingangssignal berechnet werden [GW87]. Ein lineares System ist damit komplett durch seine Impuls-Antwort charakterisiert.

Technisch wird die Impuls-Antwort oft dadurch bestimmt, daß bei einem bekannten (konstanten) Eingangssignal das Ausgangssignal gemessen wird. Aus diesem läßt sich dann h berechnen. Ein ähnlicher Vorgang ist z.B. der Weißabgleich bei Kameras.

7.2.2 Diskrete Funktion

Im diskreten Fall zweier digitalisierter Bilder unterschiedlicher Größe müssen diese auf eine einheitliche Größe erweitert werden.

$$g_e(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_e(m, n) h_e(x - m, y - n) + \eta_e(x, y)$$

Dabei sind $f(x, y)$ und $h(x, y)$ die beiden Bilder der Größe $A \times B$ und $C \times D$. Das erweiterte Bild $f_e(x, y)$ der Größe $M \times N$ ist dann definiert als

$$f_e(x, y) = \begin{cases} f(x, y) & 0 \leq x \leq A-1 \quad \text{und} \quad 0 \leq y \leq B-1 \\ 0 & A \leq x \leq M-1 \quad \text{oder} \quad B \leq y \leq N-1 \end{cases}$$

Analog ist die Erweiterung für $h_e(x, y)$ definiert. In der Matrixschreibweise lautet die letzte Gleichung

$$g = Hf + n$$

Diese vereinfachte Schreibweise wird in den folgenden Abschnitten verwendet.

Im weiteren wird diese diskrete Version behandelt, da ein Digitalrechner nur mit diskreten Werten arbeitet. Das kontinuierliche Signal kann anschließend durch Interpolation wieder angenähert werden.

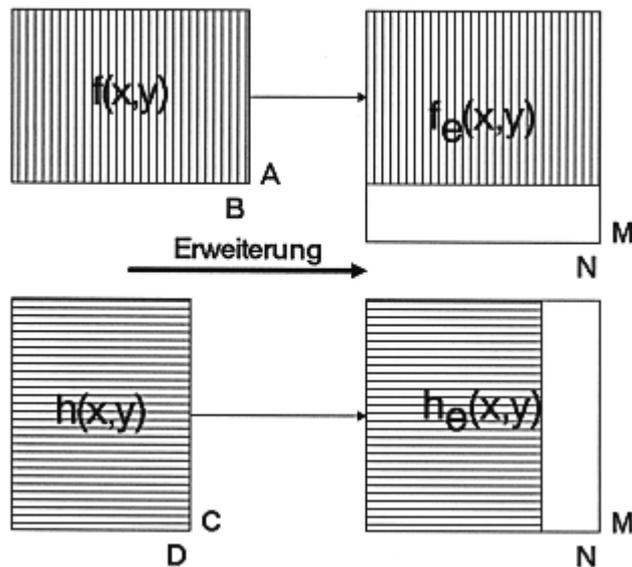


Abbildung 7.2: Die beiden Bildfunktionen werden künstlich auf eine gleiche Größe erweitert.

7.3 Inverse Filterung

Bei der *inversen Filterung* wird angenommen, daß ein Bild f durch eine Überlagerung gestört ist, deren Impuls-Antwort h bekannt ist. Diese kann man aus den physikalischen Gegebenheiten des Bildaufnahmesystems errechnen bzw. experimentell aus geeigneten Testaufnahmen abschätzen. Das Rauschen n hingegen ist ein Zufallssignal.

Man versucht daher zuerst, das restaurierte Signal optimal (im mathematischen Sinne, z.B. minimale mittlere Fehlerquadratrate) ohne Berücksichtigung des Rauschens zu berechnen. Dazu ist ein $M \times N$ -dimensionales Gleichungssystem zu lösen (falls die Determinante ungleich Null ist). Kennt man also n nicht, so ist (vorausgesetzt, die Norm von n nimmt ein Minimum an) ein \hat{f} gesucht, so daß $H\hat{f}$ das Ausgangssignal g im Sinne der kleinsten Fehlerquadratrate approximiert. D.h., es ist ein \hat{f} gesucht, so daß

$$\|n\|^2 = \|g - H\hat{f}\|^2$$

ein Minimum annimmt. Dies entspricht der Minimierung der Funktion

$$J(\hat{f}) = \|g - H\hat{f}\|^2$$

Falls H eine quadratische, invertierbare Matrix ist, ist J minimal, wenn

$$0 = g - H\hat{f}$$

Daraus erhält man dann

$$\hat{f} = H^{-1}g$$

Nicht nur aufgrund dieser großen Gleichungssysteme, sondern auch wegen der unbefriedigenden Ergebnisse schon bei nur schwach gestörten Bildsignalen wird die direkte inverse Filterung wenig eingesetzt.



Abbildung 7.3: Ein dem Originalbild in der Frequenz und der Amplitude überlagertes periodisches Störsignal (Brummen; links) kann durch die Filterung mit dem entsprechenden inversen Störsignal relativ gut restauriert werden (rechts).

7.4 Constrained-Filter

Um die Artefaktbildung der direkten inversen Filterung zu vermindern, ist die Definition von einschränkenden Randbedingungen (*constraints*) und ihre Berücksichtigung bei der Minimierung der Fehler sinnvoll.

Sei Q eine lineare Funktion auf f , so gilt es allgemein, die Funktion

$$J(\hat{f}) = \|Q\hat{f}\|^2 + \alpha(\|g - H\hat{f}\|^2 - \|n\|^2)$$

zu minimieren. Dabei ist α der sogenannte Lagrange-Multiplikator und

$$\|g - H\hat{f}\|^2 = \|n\|^2$$

die Randbedingung. Ist die Randbedingung formuliert und in die Gleichung eingebaut, so ist die Vorgehensweise, wie schon bei der inversen Filterung, vorgestellt. Man erhält

$$\hat{f} = (H^T H + \gamma Q^T Q)^{-1} H^T g$$

mit $\gamma = 1/\alpha$. Hieraus kann man leicht die Formel für die inverse Filterung erkennen. Durch die Wahl unterschiedlicher Transformationsmatrizen Q können unterschiedliche Filter abgeleitet werden, wie

7. Bildrestaurierung

z.B. der klassische Wiener-Filter. Wie leicht ersichtlich ist, ist der Erfolg dieser Constrained-Filter abhängig von der geeigneten Wahl der Randbedingungen.

7.5 Wiener-Filter

Der *Wiener-Filter* gehört zu den stochastischen Bildrestaurierungsverfahren. Diese führen eine optimale Bildverbesserung im statistischen Mittel durch. Man geht von der Vorstellung aus, daß das Originalbildsignal sowie die überlagerte Störung im Signalmodell bzw. deren korrespondierende Spektren jeweils spezielle Realisierungen zweier signalgenerierender stochastischer Prozesse sind. Beide Prozesse lassen sich mit Hilfe ihrer Korrelationsfunktion charakterisieren, die ein Maß für die statistische Bindung der Intensitätswerte benachbarter Bildpunkte darstellt. Die Korrelationsfunktion ist die Ausgangsbasis für die Matrix Q . R_f und R_n seien die Korrelationsmatrizen für das Bildsignal f und Störsignal n . Mit Hilfe des Erwartungswertes $E\{ \}$ lassen sich diese beiden Matrizen aufstellen.

$$R_f = E\{ff^T\}$$
$$R_n = E\{nn^T\}$$

Da die Elemente von f und n real sind, sind die Matrizen R_f und R_n real und symmetrisch. Bilder, deren benachbarte Bildpunkte sich intensitätsmäßig im Mittel nur geringfügig voneinander unterscheiden, besitzen langsam abfallende Korrelationsfunktionen; bei Bildszenen der natürlichen Umwelt reichen die statistischen Bindungen über 10 bis 30 Bildpunkte hinweg. Natürliche Szenen sind also relativ stark korreliert. Dadurch besitzen die Korrelationsmatrizen nur in einem Band um die Diagonale von Null verschiedene Einträge. Zufällig abrupt variierende Signale, wie beispielsweise ein den Bildern überlagertes Rauschen, haben im allgemeinen eine schnell abklingende Korrelationsfunktion. Man wählt dann für Q

$$Q^T Q = R_f^{-1} R_n$$

Das Ziel der Optimalfilterung im Wienerschen Sinn ist es, unter Ausnutzung der a-priori-Kennntnis - oder besser: der Schätzung der Korrelationsfunktionen - für das Bildsignal und für die Störung aus dem gestörten Bildsignal ein neues Signal mittels eines linearen Filters so zu berechnen, daß die quadratische Abweichung des originalen Bildsignals vom restaurierten Bildsignal im Mittel minimal wird.

Verwendet man für $\gamma = 1$, so spricht man vom klassischen Wiener-Filter, ist γ variabel, so bezeichnet man dies als das parametrisierte Wiener-Filter. Man beachte, daß für $\gamma = 1$ nicht die optimale Lösung erreicht wird, wie sie im Zusammenhang mit der inversen Filterung definiert wurde. Jedoch wird die Summe der Fehlerquadrate

$$E\{[f(x, y) - \hat{f}(x, y)]^2\} \rightarrow \min$$

im Mittel minimiert.

7.6 Interaktive Bildrestaurierung

Neben der Möglichkeit, gestörte Bilder mit Hilfe von a-priori-Information über Constrained-Filter zu restaurieren, gibt es auch die Möglichkeit, durch interaktive Arbeit eines Benutzers am Computer in die Bildrestaurierung direkt einzugreifen. So können z.B. Randbedingungen zuerst durch Betrachtung des Bildes bzw. des Fourier-Spektrums festgestellt und dann interaktiv festgelegt werden. Periodische Störungen u.ä. können somit leicht durch den Benutzer entdeckt und durch solche speziellen Filter im Frequenzraum, die nur lokal wirken (sogn. *Notch-Filter*), beseitigt werden (vergleiche Kapitel 8).

7.7 Ortsinvariante Bildrestaurierung

Die bisher besprochenen Restaurierungsverfahren waren alle ortsinvariant. Jedoch sind Fälle denkbar, wo ortsvariante Filter notwendig sind, z.B.

- Bilder mit örtlich veränderter Auflösung
- Bilder in einer ortsvariant kodierten Form
- Überlagerung des Bildsignals mit signalabhängigen Störungen usw.

Teilweise kann man diese Störungen durch ortskoordinatenabhängige Filter (geometrische Transformationen wie Skalierung, Rotation und Koordinatentransformation) oder durch Filter, deren Übertragungsverhalten von den zu filternden Bildsignalen selbst abhängig ist, wieder beseitigen.

7. Bildrestaurierung

8. Fourier-Transformation

Funktionen reeller Variablen lassen sich oft in Form von Reihen darstellen. Dabei spielen die nach dem französischen Ägyptologen und Naturwissenschaftler Jean Baptiste Joseph Fourier (1768-1830) benannten Reihen eine besondere Rolle. Er erkannte 1822, daß solche Funktionen als eine Überlagerung (*Superposition*) von harmonischen Schwingungen unterschiedlicher Frequenz und Amplitude aufgefaßt werden können. Im einfachsten Fall sind dies Sinus- und Kosinus-Schwingungen.

Eine eindimensionale periodische Funktion $f(x)$ mit der Periode $T = 2\pi$ läßt sich damit folgendermaßen darstellen:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos(kx) + b_k \sin(kx))$$

Die endliche *Fourier-Transformation* ordnet dieser Funktion aus dem Zeitbereich bestimmte Koeffizienten zu, die die Amplitude der einzelnen harmonischen Schwingungen darstellen. Diese Fourier-Koeffizienten a_k, b_k ergeben sich aus

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(kx) dx, \quad k = 0, 1, 2, \dots \quad \pi = \frac{T}{2}$$

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(kx) dx, \quad k = 0, 1, 2, \dots \quad \pi = \frac{T}{2}$$

Die Fourier-Transformierte im Frequenzbereich enthält demnach exakt die gleiche Information wie das zugehörige Signal im Zeitbereich. Die beiden unterscheiden sich nur in der Darstellung.

Diese anschauliche Darstellung durch bekannte trigonometrische Funktionen wird leider völlig unübersichtlich, wenn mit Fourierreihen gerechnet werden soll. Es ist daher viel einfacher, für die Berechnungen komplexwertige Exponentialschwingungen zu verwenden. Der Zusammenhang ist gegeben durch die Euler-Identität:

$$e^{ikx} = \cos(kx) + i \sin(kx)$$

Hierbei handelt es sich um komplexe Zahlen auf dem Einheitskreis (mit dem Radius 1). Interessant sind hier nun die k -ten Einheitswurzeln $W_k = e^{(2\pi i/k)}$, die k -mal mit sich selbst multipliziert 1 ergeben.

Man erhält damit die äquivalente Darstellung

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx}, \quad \text{mit} \quad c_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) e^{-ikx} dx = \begin{cases} \frac{a_0}{2} & k = 0 \\ \frac{1}{2}(a_k - i b_k) & k > 0 \\ \frac{1}{2}(a_{-k} + i b_{-k}) & k < 0 \end{cases}$$

8. Fourier-Transformation

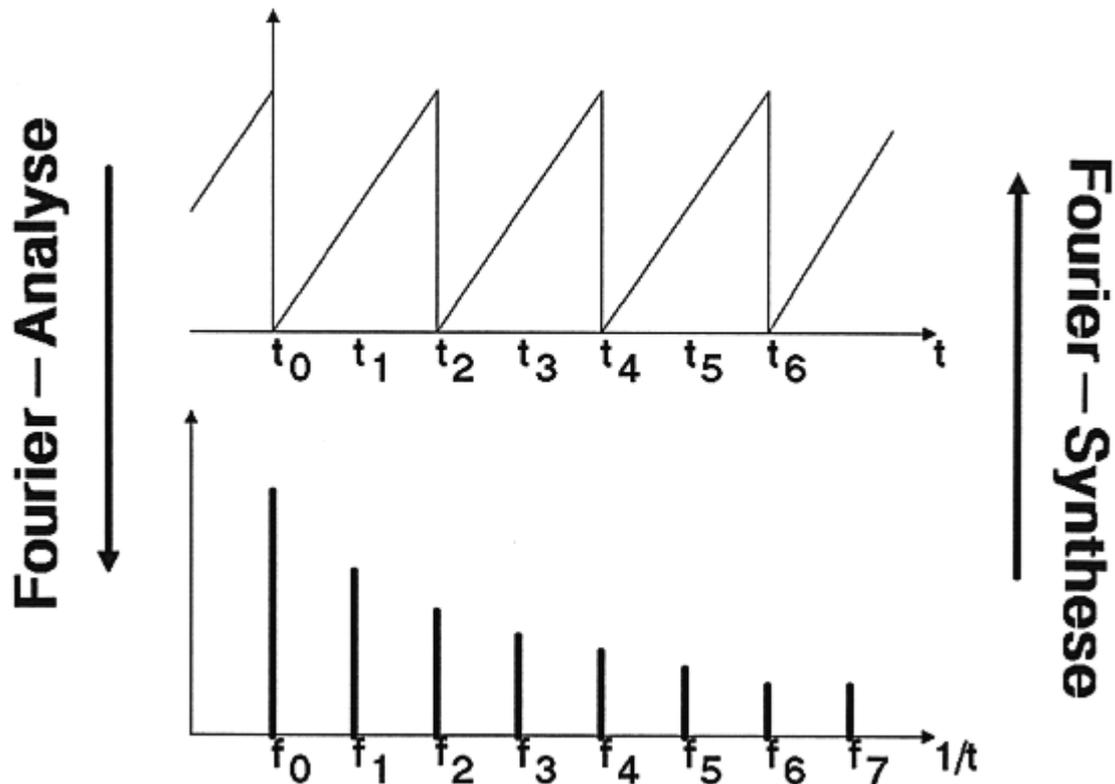


Abbildung 8.1: Die Fourier-Analyse einer sägezahnförmigen Zeitfunktion ergibt das Spektrum dieser Funktion, d.h., die einzelnen Frequenzen mit ihren unterschiedlichen Amplituden, die in der Summe die Sägezahnfunktion ergeben. Der umgekehrte Vorgang, die Fourier-Synthese, liefert den zeitlichen Verlauf zurück.

Die c_k sind komplexwertig und stellen das diskrete Frequenzspektrum der Funktion $f(x)$ dar. Man nennt sie *Fourier-Koeffizienten*. Ist die Funktion $f(x)$ periodisch mit der Periode $2l$, so gestattet sie die Entwicklung

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos\left(k \frac{\pi}{l} x\right) + b_k \sin\left(k \frac{\pi}{l} x\right) \right)$$

oder in einer anderen Schreibweise

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx(\pi/l)}, \quad \text{mit} \quad c_k = \frac{1}{2l} \int_0^{2l} f(x) e^{-ikx(\pi/l)} dx$$

Es liegt nun nahe, in der Fourier-Reihe einer im Intervall $-l < x < l$ gegebenen Funktion $f(x)$ den Grenzübergang $l \Rightarrow \infty$ zu versuchen, um sich von dem Zwang der periodischen Fortsetzung von $f(x)$ zu befreien und eine Darstellung für alle reellen x definierten, nichtperiodischen Funktionen zu gewinnen. Man erhält dann

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} du \int_{-\infty}^{\infty} f(t) e^{-iu(t-x)} dt$$

Diese Gleichung läßt sich als die Superposition der folgenden Formeln auffassen:

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-iux} dx$$

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(u) e^{iux} du$$

Man sagt, die Fourier-Transformierte $F(u)$ liefert eine Entwicklung der Funktion $f(x)$ in ein kontinuierliches Spektrum (*Ortsfrequenzspektrum*). Der Parameter u wird in Analogie zur Frequenz bei Zeitfunktionen *Ortsfrequenz* genannt.

Die Wahl der Faktoren vor den beiden Integralen (1 und $1/2\pi$) wird in der Literatur unterschiedlich vorgenommen, je nachdem, wie der Zusammenhang zwischen der Laplace-Transformation und der Fourier-Transformation sowie der Energie eines Signals im Zeitbereich und seiner Energie im Frequenzbereich definiert ist. Das Produkt der beiden Faktoren muß auf jeden Fall $1/2\pi$ ergeben. Eine umfassende Erklärung ist in [Bri89] zu finden.

Kennt man den Verlauf einer Funktion f in Abhängigkeit z.B. der Zeit t , so kann man die Frequenzen f berechnen, die den Verlauf der zeitlichen Funktion ausmachen – man bestimmt das Frequenzspektrum (*Fourier-Analyse*). Der umgekehrte Weg ist ebenfalls möglich. Die Konstruktion einer Funktion der Zeit aus ihrem Frequenzspektrum nennt man *Fourier-Synthese* (siehe Abbildung 8.1).

Im 2-dimensionalen Fall von Bildern (jedem Bildpunkt (x, y) wird eine Intensität oder Farbe $I(x, y)$ zugeordnet und dieser Wert $I(x, y)$ als abgetasteter Amplitudenwert einer kontinuierlichen Funktion angesehen) erhält man:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i(ux+vy)} dx dy$$

$$f(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{i(ux+vy)} du dv$$

Bei diesen Formeln werden Zeitfunktionen mit Kleinbuchstaben, die Fourier-Transformierten mit den entsprechenden Großbuchstaben bezeichnet.

Da die Fourier-Transformierte einer reellen Funktion im allgemeinen komplexwertig ist, kann man auch schreiben $F(u, v) = R(u, v) + i I(u, v)$, wobei $R(u, v)$ und $I(u, v)$ der Real- bzw. Imaginärteil von $F(u, v)$ sind. Der Betrag von $F(u, v)$ ist

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}$$

und wird als *Fourier-Spektrum* (auch Amplitudenspektrum) bezeichnet.

Bei der Visualisierung dieser Daten wird die Amplitude des Fourier-Spektrums in der Regel durch Grauwerte (hell=hohe Amplitude) dargestellt. Zur besseren Veranschaulichung sind die

8. Fourier-Transformation

Werte oft noch logarithmisch abgestuft. Das *Leistungsspektrum* ist das Quadrat des Fourier-Spektrums. Der Ausdruck

$$\Phi(u, v) = \arctan\left(\frac{I(u, v)}{R(u, v)}\right)$$

wird als das Phasenspektrum bezeichnet.

Oft wird $F(u, v)$ auch als Produkt von Amplitudenspektrum $|F(u, v)|$ und Phasenspektrum $e^{i\Phi(u, v)}$ geschrieben:

$$F(u, v) = |F(u, v)| e^{i\Phi(u, v)}$$

Eine für die folgenden Betrachtungen wichtige Gruppe stellen die Fourier-Transformierten von singulären Funktionen (*Delta-Funktionen* $\delta(x)$) dar. Die Delta-Funktion $\delta(x)$ ist definiert als

$$\delta(x) = \begin{cases} 1 & \text{für } x=0 \\ 0 & \text{sonst} \end{cases}$$

Damit gilt

$$f(x_0) = \int_{-\infty}^{\infty} \delta(x - x_0) f(x) dx$$

wobei $f(x)$ eine beliebige und bei x_0 stetige Funktion ist. Für die Funktion $f(x) = K \delta(x)$ (K eine beliebige Konstante) berechnet sich die Fourier-Transformierte nach

$$F(x) = \int_{-\infty}^{\infty} K \delta(x) e^{-iux} dx = Ke^0 = K$$

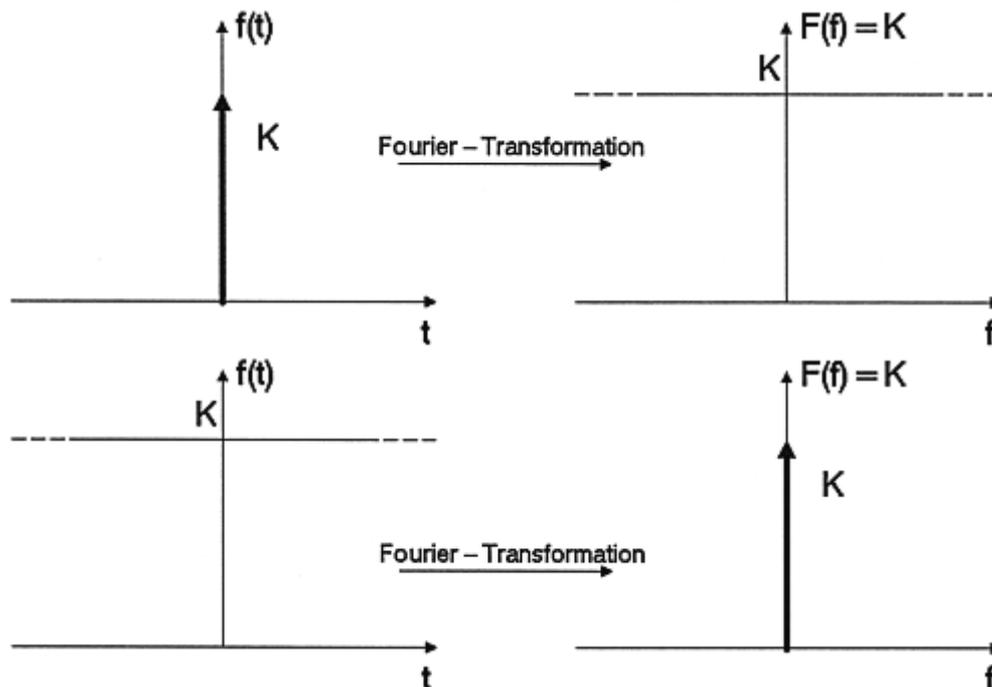


Abbildung 8.2: Die Delta-Funktion und ihre Fourier-Transformierte (oben) sowie eine konstante Funktion mit ihrer Fourier-Transformierten (unten).

8.1 Diskrete Fourier-Transformation (DFT)

$F(u)$ und $f(x)$ sind kontinuierliche Funktionen. Für die Anwendungen in der Bildverarbeitung ist diese kontinuierliche Form der Fourier-Transformation natürlich ungeeignet. Man verwendet daher die *diskrete Fourier-Transformation* (DFT) mit

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-i2\pi ux/N} \quad u = 0, 1, \dots, N-1$$

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{i2\pi ux/N} \quad x = 0, 1, \dots, N-1$$

Die Werte von u entsprechen in der diskreten Fourier-Transformation den Abtastungen der kontinuierlichen Funktion an den Stellen $0, \Delta u, 2\Delta u, \dots$. Anwendung der diskreten Fourier-Transformation auf eine Funktion, die z.B. einen zeitlichen Verlauf kennzeichnet, liefert die komplexen Spektrallinien $F(u)$, die zu den diskreten Frequenzen $t(u/N)$ gehören. Je dichter man die "Meßwerte" $f(x)$ legt, desto besser wird die tatsächliche Funktion approximiert. Üblicherweise spricht man nicht von Meßwerten, sondern von Stützstellen für die Approximation. Sind viele Stützstellen bekannt, können die Frequenzen, die auftreten, gut bestimmt werden. Die Anzahl N der Stützstellen entspricht bei digitalen Bildern der Auflösung.

Der Term $K_u = e^{-i2\pi ux/N}$ wird als Kern K der diskreten Fourier-Transformation bezeichnet. Er stellt die Basisfunktionen dar (vergleiche Kapitel 9). Diese Basisfunktionen sind nach der Euler-Identität Sinus- bzw. Kosinuswellen mit verschiedener Wellenlänge, wobei der Index u (Wellenzahlindex) angibt, wie oft die Periode der Basisfunktion K_u in das Intervall von $0, \dots, N-1$ hineinpaßt.

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-i2\pi(ux/M + vy/N)} \quad u = 0, 1, \dots, M-1$$

$$v = 0, 1, \dots, N-1$$

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{i2\pi(ux/M + vy/N)} \quad x = 0, 1, \dots, M-1$$

$$y = 0, 1, \dots, N-1$$

```
void DFT_1Dim(int BildNr, int Zeilen_Richtung, int Nummer)
/* BildNr = Nummer des Bildes von dem Zeilen/Spalten transform. */
/* Zeilen_Richtung = Flag, ob in Zeilen- oder Spaltenrichtung */
/* Nummer = Nummer der zu transformierenden Zeile/Spalte */
{
  int Punkt_Zahl;          /* zu transformierende Anzahl Pkt. */
  int z, s;                /* Zeilen/Spalten-Index */
  int i, j;                /* Index-Variablen */
}
```

8. Fourier-Transformation

```
float Trafo_Werte[RES_MAX]; /* Vektor der zu transform. Werte */
float Amplitude[RES_MAX]; /* Werte des Amplitudenspektrums */
float Phase[RES_MAX]; /* Werte des Phasenspektrums */
float Real_Teil, Imag_Teil; /* Hilfsvariablen Real-/Imag.-Teil */
float Phi, Phi_Inkr; /* Winkel und Inkrement */

ClearTextWindow(35,18,80,25);
WriteText(35,18,"* DFT - eindimensional *");

/* zu transformierende Werte in einen Hilfs-Vektor kopieren */
if (Zeilen_Richtung) { /* eine Zeile transformieren */
    Punkt_Zahl = Picture[BildNr].Spalten;
    for (s=0; s<Punkt_Zahl; s++)
        Trafo_Werte[s] = (float)Picture[BildNr].Bild[Nummer][s];
}
else { /* eine Spalte transformieren */
    Punkt_Zahl = Picture[BildNr].Zeilen;
    for (z=0; z<Punkt_Zahl; z++)
        Trafo_Werte[z] = (float)Picture[BildNr].Bild[z][Nummer];
}
/* Jetzt die DFT dieses Float-Hilfs-Vektors berechnen */
for (i=0; i<Punkt_Zahl; i++) {
    /* Initialisierung der Variablen */
    Real_Teil=0;
    Imag_Teil=0;
    Phi =0;
    Phi_Inkr = 2.0*PI*i/(float)(Punkt_Zahl);
    /* Transformation streng nach Definition vornehmen */
    for (j=0; j<Punkt_Zahl; j++) {
        Real_Teil +=Trafo_Werte[j] * cos(Phi);
        Imag_Teil +=Trafo_Werte[j] * sin(Phi);
        Phi -=Phi_Inkr;
    }
    /* Amplitudenspektrum und Phasenspektrum berechnen */
    Amplitude[i]=sqrt(Real_Teil*Real_Teil + Imag_Teil*Imag_Teil);
    Phase[i] =atan2(Real_Teil, Imag_Teil);
}
/* Die Ergebnisse koennen jetzt als Histogramm angezeigt werden*/
}
```

8.2 Eigenschaften der Fourier-Transformation

Wie schon erwahnt enthalt die Darstellung eines Signals im Ortsraum dieselbe Information wie die Darstellung als Fourier-Transformierte im Frequenzraum. Verschiedene Berechnungen lassen sich im Frequenzraum aber leichter durchfuhren, bestimmte Eigenschaften besser ablesen. Die wichtigsten Eigenschaften der Fourier-Transformationen sollen daher hier noch einmal kurz aufgelistet werden. Diese gelten sowohl fur den diskreten als auch fur den kontinuierlichen ein- oder zweidimensionalen Fall.

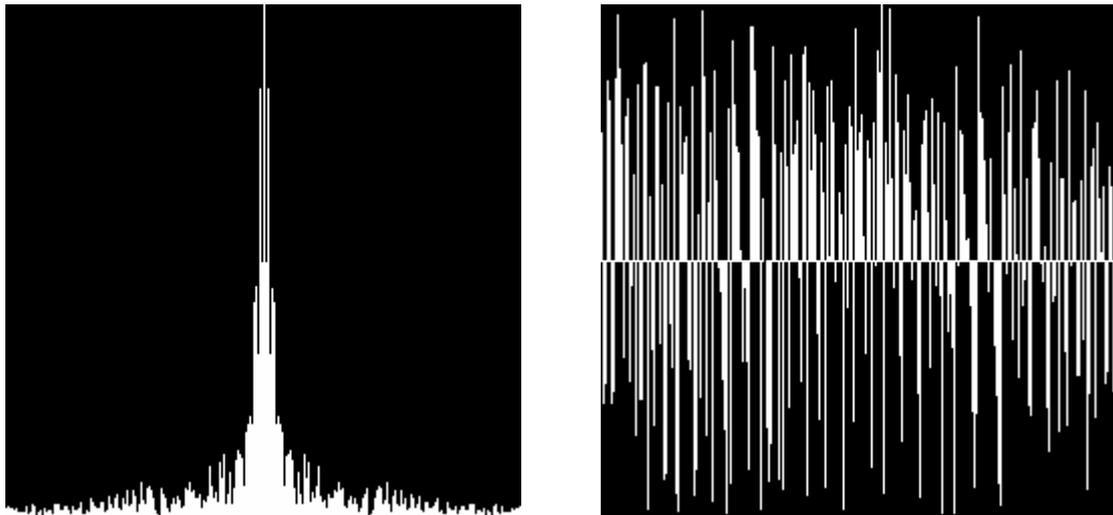


Abbildung 8.3: Die Fourier-Transformation einer Zeile aus einem Grauwertbild liefert das Amplitudenspektrum (links) und das Phasenspektrum (rechts). Eine logarithmische Gewichtung der Werte wurde hier nicht vorgenommen.

- **Linearität, Überlagerung**

Die Fourier-Transformierte von Summen von Ortsfunktionen ist gleich der Summe ihrer Fourier-Transformierten, d.h., wenn $F(u)$ und $G(u)$ die Fourier-Transformierten von $f(x)$ und $g(x)$ sind, dann ist die Fourier-Transformierte der Summe $f(x) + g(x)$ gleich $F(u) + G(u)$.

- **Separierbarkeit**

Die zweidimensionale Fourier-Transformation kann mit Hilfe einer eindimensionalen Fourier-Transformation zunächst nach der einen Variablen und anschließend einer weiteren eindimensionalen Fourier-Transformation der so erhaltenen Zwischengröße nach der anderen Variablen berechnet werden.

- **Periodizität**

Die Fourier-Transformierte einer im Intervall $(0 \leq m \leq M-1; 0 \leq n \leq N-1)$ definierten diskreten Funktion entspricht der um den Faktor M bzw. N verschobenen Fourier-Transformierten, d.h. $F(u, v) = F(u+M, v) = F(u, v+N) = F(u+M, v+N)$.

- **Symmetrie**

Symmetrien unter der Fourier-Transformation bleiben erhalten, d.h. eine gerade bzw. ungerade Funktion besitzt ein gerades bzw. ungerades Spektrum. Analog gilt daher auch für eine diskrete Funktion $F(u, v) = \pm F(-u, -v)$ mit $+$ für gerade und $-$ für ungerade Funktionen. Unter Ausnutzung der Periodizität findet man die negativen, außerhalb der Matrix liegenden Werte

8. Fourier-Transformation

in der Matrix wieder, so daß $F(u, v) = \pm F(M-u, N-v)$ gilt. Der Punkt $(M/2, N/2)$ ist das Symmetriezentrum.

- **Zerlegung**

Eine Funktion kann in einen geraden und einen ungeraden Anteil zerlegt werden. Die Fourier-Transformierte einer geraden Funktion ist eine gerade Funktion, die einer ungeraden eine ungerade imaginäre Funktion (vergleiche: Symmetrie). Eine beliebige Funktion $f(x)$ läßt sich schreiben als

$$f(x) = \frac{f(x)}{2} + \frac{f(x)}{2}$$
$$f(x) = \left[\frac{f(x)}{2} + \frac{f(-x)}{2} \right] + \left[\frac{f(x)}{2} - \frac{f(-x)}{2} \right]$$
$$f(x) = f_g(x) + f_u(x)$$

Für die Fourier-Transformation ergibt sich dann

$$F(u) = R(u) + i I(u) = F_g(u) + F_u(u)$$

mit $F_g(u) = R(u)$ und $F_u(u) = i I(u)$.

- **Vertauschung**

Faßt man das Ortsfrequenzspektrum $F(u, v)$ der Ortsfunktion $f(x, y)$ als eine neue Ortsfunktion $F(x, y)$ auf, so hat deren Fourier-Transformierte die an den Koordinatenachsen gespiegelte Form $f(-u, -v)$ des ursprünglichen Signals, multipliziert mit $4\pi^2$.

- **Skalierung, Ähnlichkeit**

Eine Dehnung der Ortsfunktion $f(x, y)$ in der Ortsebene führt zu einer Skalierung und Stauchung von $F(u, v)$ in der Ortsfrequenzebene und umgekehrt, d.h., die Fourier-Transformierte von $f(ax, by)$ ist

$$\frac{1}{\|ab\|} F\left(\frac{u}{a}, \frac{v}{b}\right).$$

- **Rotation**

Eine Drehung der Funktion $f(x, y)$ im Ortsbereich um einen Winkel α bewirkt eine Drehung von $F(u, v)$ in der Ortsfrequenzebene um den gleichen Winkel im gleichen Richtungssinn und umgekehrt (siehe Abbildung 8.5).

- **Verschiebung**

Eine Verschiebung der Funktion $f(x, y)$ im Ortsbereich bewirkt eine lineare Phasendrehung der Funktion $F(u, v)$ im Ortsfrequenzbereich und umgekehrt. Aus diesem Satz wird auch die Bedeutung der Phase klar. Sie legt die Position und die Struktur der Objekte im Ortsraum fest.

- **Faltung**

Die Faltung (*) zweier Funktionen $f(x)$ und $g(x)$ ist definiert als

$$h(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(\alpha) g(x-\alpha) d\alpha$$

Die Fourier-Transformierte des Faltungsprodukts $h(x)$ ist gleich dem Produkt der Fourier-Transformierten der einzelnen Funktionen. Damit ist ein wichtiges Einsatzgebiet der Fourier-Transformation angesprochen. Eine rechenintensive Faltung (Integration) im Ortsraum kann durch eine einfache Multiplikation im Ortsfrequenzraum durchgeführt werden.

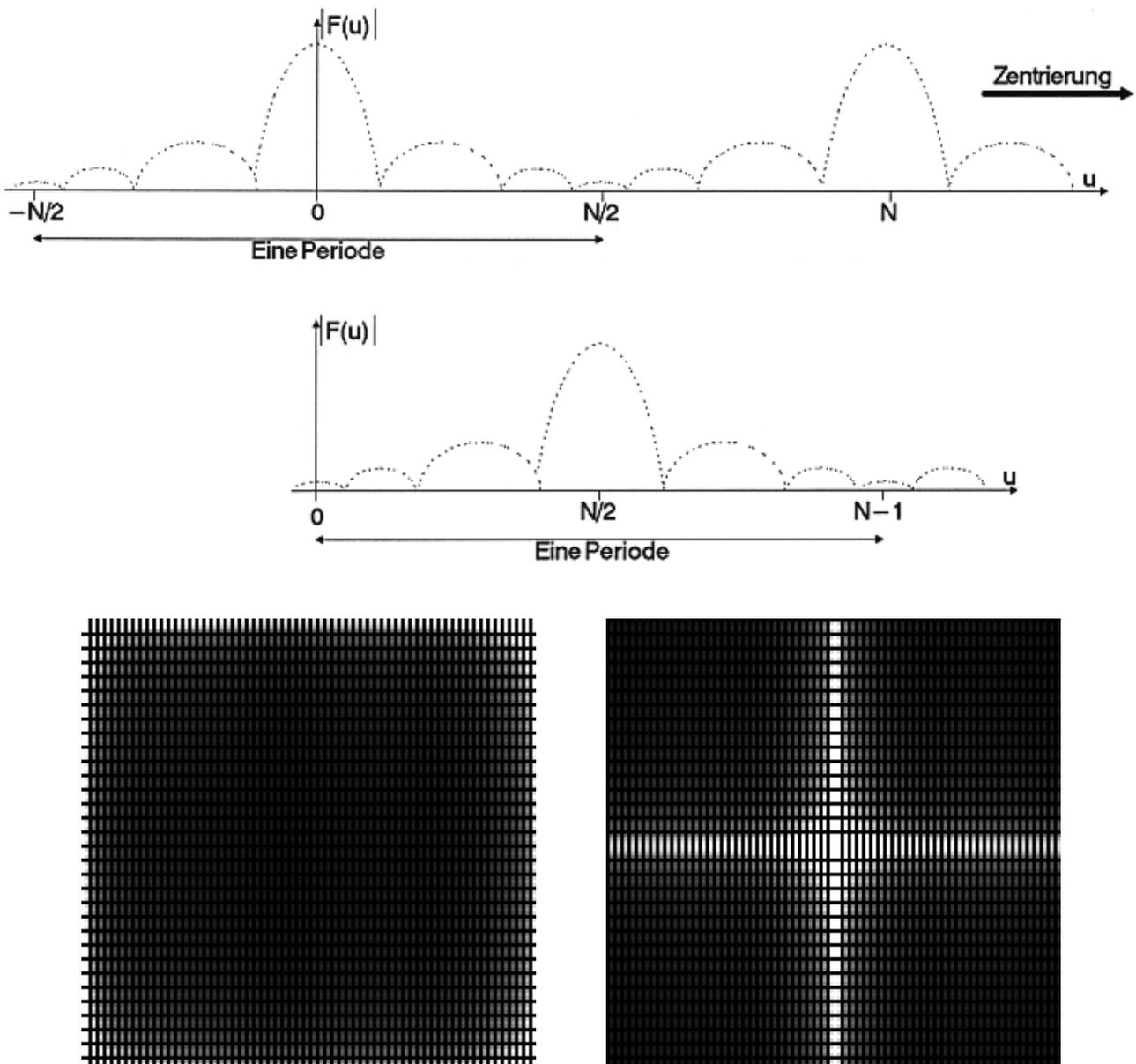


Abbildung 8.4: Punktsymmetrie der Fourier-Transformation im Amplitudenspektrum. Bei der Darstellung des Amplitudenspektrums werden die Amplituden der tiefen Frequenzen in den Bildmittelpunkt verschoben. Oben ist das Original-Amplitudenspektrum einer 1D-Transformation, darunter das zentrierte Amplitudenspektrum zu sehen. Unten links ist analog das Spektrum einer 2D-Transformation und rechts davon die gewohnte im Bildmittelpunkt zentrierte Darstellung.

8. Fourier-Transformation

- **Korrelation**

Die Korrelation (\circ) zweier Funktionen $f(x)$ und $g(x)$ ist definiert als

$$h(x) = f(x) \circ g(x) = \int_{-\infty}^{\infty} f^*(\alpha) g(x+\alpha) d\alpha$$

wobei $*$ die konjugiert komplexe Funktion kennzeichnet. Die Korrelation wird benötigt, um Ähnlichkeiten von Bildern/Objekten (Mustervergleich) herauszufinden (siehe Abschnitt 9.5). Die Suche nach dem größten Wert im zweidimensionalen Raum kann dann in der Regel durch eine Berechnung im Ortsfrequenzraum schneller erfolgen.

Zur graphischen Darstellung der Ergebnisse der Fourier-Transformation wird die Zahlenmatrix so aufbereitet, daß die Punktsymmetrie der Fourier-Transformierten Funktion deutlich wird (in einer bestimmten Richtung auftretende Frequenzen findet man gleichermaßen in der entgegengesetzten Richtung wieder). Die Darstellung erfolgt dann über das Fourier-Spektrum. Da diese Werte mit zunehmender Frequenz sehr stark abnehmen, wird oft eine logarithmische Darstellungsform verwendet.

$$D(u, v) = c \log(1 + |F(u, v)|)$$

Auf dem Bildschirm wird das Fourier-Spektrum als Grauwertbild ausgegeben. Üblicherweise werden große Werte hell, kleinere dunkel dargestellt. Dabei wird eine Normierung der Grauwerte vorgenommen, um genau den zur Verfügung stehenden Grauwertbereich auszunutzen.

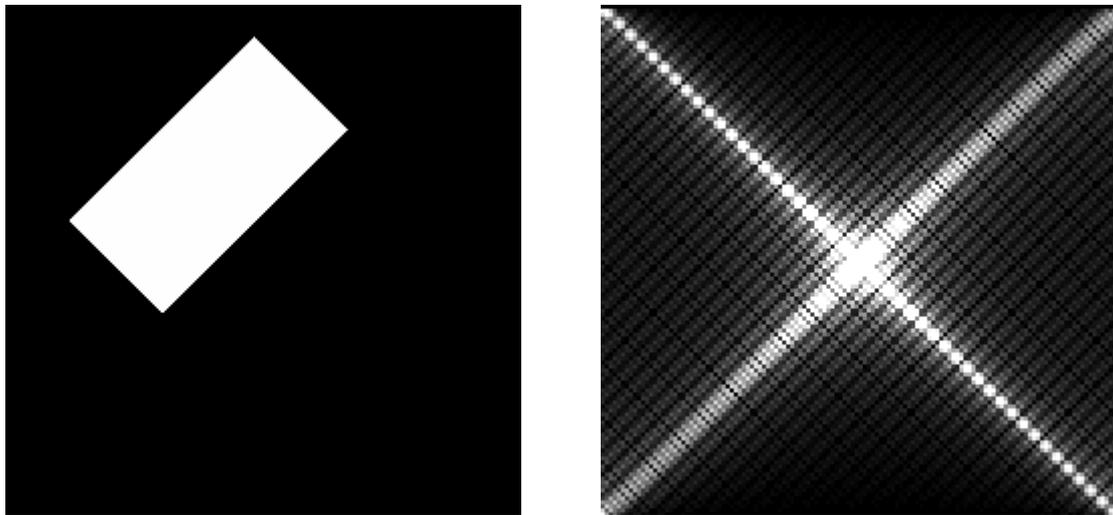


Abbildung 8.5: Eine Rotation der Funktion (Objekt) im Ortsbereich bewirkt auch eine Rotation des Fourier-Spektrums im Ortsfrequenzbereich.

Prinzipiell gilt die endliche Fourier-Transformation ja nur für periodische Signale oder Funktionen, so daß die Festlegung auf die Bildgröße nicht trivial ist. Es hat sich jedoch als sinnvoll erwiesen, die Grenzen für die Fourier-Transformation gleich den sichtbaren Bildgrenzen zu setzen. Andere Mög-

lichkeiten bestünden z.B. darin, einen schwarzen Rand um das Bild zu legen oder Filter zu benutzen, die die Bildränder weniger stark berücksichtigen.

Aus den oben genannten Eigenschaften lassen sich einzelne Muster im Fourier-Spektrum entsprechend deuten. So führen z.B. Verwacklungen des Originalbildes im Fourier-Spektrum zu zur Verwacklungsrichtung orthogonalen Streifen, eine Drehung der Ortsfunktion zu einer Drehung des Fourier-Spektrums um den gleichen Winkel und in die gleiche Richtung. Bei Unschärfe entstehen konzentrische Kreise um den Mittelpunkt. Hochfrequente Anteile eines Bildes geben Hinweise auf mögliche Kanten, so daß mit der Fourier-Transformation auch Kantendetektionsverfahren entwickelt werden können.

8.3 Fast-Fourier-Transformation (FFT)

Aufgrund der großen Anzahl von Multiplikationen und Additionen bei der normalen Fourier-Transformation wurde und wird immer wieder versucht, dieses Verfahren zu optimieren. Kriterien dazu sind vor allem Schnelligkeit, Einfachheit und Speicherplatzverwendung. 1965 wurde von Cooley und Tukey der FFT-Algorithmus (*Fast-Fourier-Transformation*) zum erstenmal veröffentlicht.

Die Geschwindigkeitssteigerung wird im wesentlichen durch eine bestimmte Anzahl von Bildpunkten bzw. Stützwerten (diese muß eine Zweierpotenz sein; in den letzten Jahren sind einige neue, zum Teil sehr komplizierte FFT-Verfahren beschrieben worden, die zwar nur geringfügig schneller sind als das von Cooley und Tukey, jedoch andere Punktzahlen erlauben) und durch Vorsortieren der Werte erreicht. Die FFT ist also keine Näherung der diskreten Fourier-Transformation, sondern liefert dieselben Ergebnisse.

Der FFT-Algorithmus nutzt Symmetrieeigenschaften einer DFT mit $N = 2^k$ aus und reduziert die Anzahl der erforderlichen Multiplikationen durch Ausklammern von Faktoren. Die Vorgehensweise der FFT soll anhand einer eindimensionalen Funktion $f(n)$ erklärt werden. Diese ist in einem Intervall $\{0 \leq n \leq N-1\}$ definiert, wobei N eine Zweierpotenz ist. Die eindimensionale diskrete Fourier-Transformation von $f(n)$ lautet dann

$$F(u) = \frac{1}{N} \sum_{n=0}^{N-1} f(n) e^{-i2\pi un/N} \quad 0 \leq u \leq N-1$$

Seien die Einheitswurzeln wieder mit $W_N = e^{-i2\pi/N}$ bezeichnet (oft wird N als Index zu W zur besseren Übersichtlichkeit weggelassen)

$$F(u) = \frac{1}{N} \sum_{n=0}^{N-1} f(n) W_N^{un} \quad 0 \leq u \leq N-1$$

so ergibt sich beispielsweise für $N = 4$ das Gleichungssystem

8. Fourier-Transformation

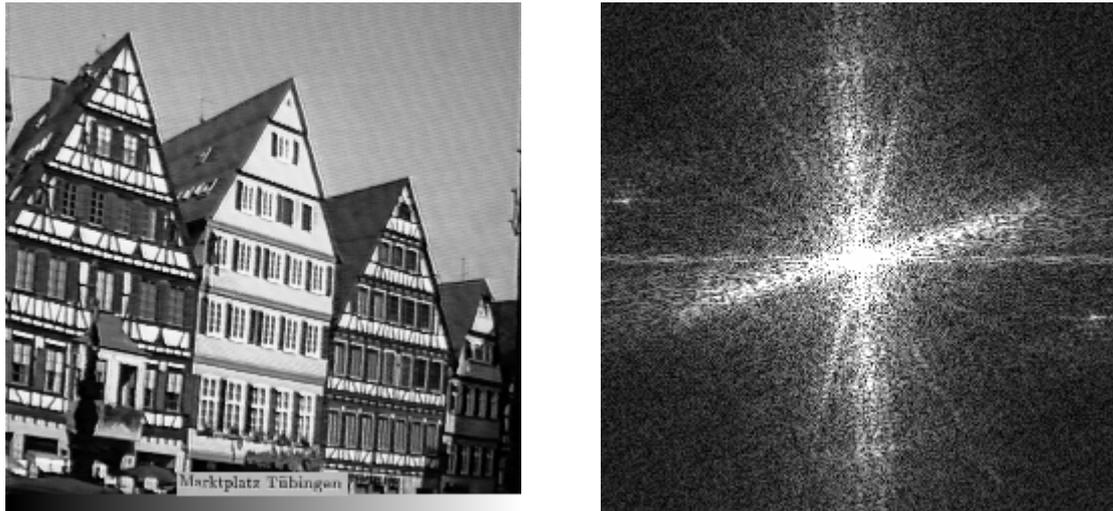


Abbildung 8.6: Das Amplitudenspektrum (rechts) der Fourier-Transformierten des linken Bildes. Das Symmetriezentrum ist hier wieder in die Bildmitte gelegt.

$$\begin{aligned}
 F(0) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^0 + f(2)W_4^0 + f(3)W_4^0) \\
 F(1) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^1 + f(2)W_4^2 + f(3)W_4^3) \\
 F(2) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^2 + f(2)W_4^4 + f(3)W_4^6) \\
 F(3) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^3 + f(2)W_4^6 + f(3)W_4^9)
 \end{aligned}$$

Diese Gleichungen lassen sich in einer Matrixform einfacher schreiben. Aufgrund der Periodizität auf dem Einheitskreis gilt die Beziehung $W_N^{nu} = W^{nu \bmod N}$. Im Beispiel mit $N = 4$ ergibt sich damit $W_4^6 = W_4^2$ usw. Das obige Gleichungssystem kann dann äquivalent in der Form

$$\begin{aligned}
 F(0) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^0 + f(2)W_4^0 + f(3)W_4^0) \\
 F(1) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^1 + f(2)W_4^2 + f(3)W_4^3) \\
 F(2) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^2 + f(2)W_4^0 + f(3)W_4^2) \\
 F(3) &= \frac{1}{4}(f(0)W_4^0 + f(1)W_4^3 + f(2)W_4^2 + f(3)W_4^1)
 \end{aligned}$$

geschrieben werden.

Wie man an diesem Gleichungssystem erkennt, sind für die Berechnung von $F(u)$ normalerweise N^2 komplexe Additionen und Multiplikationen notwendig. Bei der schnellen Fourier-Transformation wird nun die ursprüngliche Funktion $f(n)$ in zwei Teilsequenzen der Länge $M = N/2$ zerlegt, diese wieder aufgeteilt usw. bis man schließlich zur diskreten Fourier-Transformation von jeweils zwei

Elementen gelangt (*divide-and-conquer*). Jetzt kann man schrittweise die nächst höhere Stufe berechnen. Für $N = 2M$ gilt

$$F(u) = \frac{1}{2M} \sum_{n=0}^{2M-1} f(n) e^{(-i2\pi un)/2M}$$

Aus

$$F(u) = \frac{1}{2M} \sum_{n=0}^{M-1} f(2n) W_{2M}^{u(2n)} + \frac{1}{2M} \sum_{n=0}^{M-1} f(2n+1) W_{2M}^{u(2n+1)}$$

ergibt sich dann wegen $W_{2M}^{2nu} = W_M^{nu}$

$$F(u) = \frac{1}{2} \left(\frac{1}{M} \sum_{n=0}^{M-1} f(2n) W_M^{un} + \frac{1}{M} \sum_{n=0}^{M-1} f(2n+1) W_M^{un} W_{2M}^u \right)$$

Definiert man den geraden Anteil $F_g(u)$ und den ungeraden Anteil $F_u(u)$ der Fourier-Transformierten $F(u)$ mit

$$F_g(u) = \frac{1}{M} \sum_{n=0}^{M-1} f(2n) W_M^{un} \quad \text{und} \quad F_u(u) = \frac{1}{M} \sum_{n=0}^{M-1} f(2n+1) W_M^{un}$$

so vereinfacht sich die Gleichung in

$$F(u) = \frac{1}{2} (F_g(u) + F_u(u) W_{2M}^u)$$

Zusätzlich gilt $W_M^{u+M} = W_M^u$ und $W_{2M}^{u+M} = -W_{2M}^u$ sowie $W_{2M}^M = -W_{2M}^0 = -1$. Aus dieser Symmetrieeigenschaft folgt damit

$$F(u+M) = \frac{1}{2} (F_g(u) - F_u(u) W_{2M}^u)$$

Durch diese Unterteilung und Art der Berechnung sind nur noch $\frac{N}{2} \log_2 N$ komplexe Multiplikationen notwendig. Die FFT besteht nur noch aus $\log_2 N$ Stufen; der Speicherbereich der ursprünglichen Datensequenz kann jeweils mit den Zwischenergebnissen der zuletzt berechneten Stufe überschrieben werden.

Wegen der Periodizität und der Symmetrieeigenschaften der Fourier-Transformation sind die Summanden von $F(0)$ und $F(2)$ bzw. $F(1)$ und $F(3)$ bis auf das Vorzeichen identisch. Im Beispiel ergibt sich damit

$$\begin{aligned} F(0) &= \frac{1}{2} \left(\left(\frac{1}{2} f(0) W_2^0 + \frac{1}{2} f(2) W_2^0 \right) + \left(\frac{1}{2} f(1) W_2^0 + \frac{1}{2} f(3) W_2^0 \right) W_4^0 \right) \\ &= \frac{1}{4} (f(0) + f(2) + f(1) + f(3)) \\ F(1) &= \frac{1}{2} \left(\left(\frac{1}{2} f(0) W_2^0 - \frac{1}{2} f(2) W_2^1 \right) + \left(\frac{1}{2} f(1) W_2^0 - \frac{1}{2} f(3) W_2^1 \right) W_4^1 \right) \\ &= \frac{1}{4} (f(0) - f(2) W_2^1) + f(1) - f(3) W_2^1 W_4^1 \\ &= \frac{1}{4} (f(0) - f(2) + f(1) - f(3) W_4^1) \end{aligned}$$

8. Fourier-Transformation

$$\begin{aligned}
 F(2) &= \frac{1}{4}(f(0) & + f(2) & - (f(1) & + f(3))) \\
 F(3) &= \frac{1}{4}(f(0) & - f(2) & - (f(1) & - f(3)) W_4^1)
 \end{aligned}$$

Durch die Ausnutzung der Eigenschaften der Fourier-Transformation und der mehrfachen Verwendung von Zwischenergebnissen kann der Aufwand an Multiplikationen und Additionen bei der FFT gegenüber der normalen DFT erheblich gesenkt werden. Auf den ersten Blick etwas unverständlich erscheint die Verknüpfung der einzelnen Funktionswerte und die Multiplikation mit den entsprechenden W_i^n . Durch eine Vorsortierung wird diese Vorgehensweise aber vereinfacht. Diese Vorsortierung wird dadurch erreicht, daß die Indexpzahlen in ihrer Binärdarstellung an der Mitte gespiegelt oder umgekehrt gelesen werden. Aus 4=100 wird dann 001=1. Diese neuen Zahlen mit ihrem Dezimalwert werden dann als Indexpzahlen für die FFT verwendet. Dieser Vorgang wird mit *Bit-Reverse-Shuffling* bezeichnet.

Original		Sortiert	
Dezimal	Binär	Binär	Dezimal
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

Tabelle 8.1: Die Indizes für die Funktionswerte werden durch Spiegelung der Zahlen in der Binärdarstellung an der Mittelachse erzeugt (Bit-Reverse-Shuffling). Das Beispiel für $N = 8$ Ausgangswerte verdeutlicht dies.

Die komplette Verknüpfung der Funktionswerte mit den W_i^n wird bei wenigen Elementen auch in einem sogenannten *Signalflußgraphen* dargestellt (siehe Abbildung 8.7). Begonnen wird bei der Berechnung mit den sortierten Ausgangswerten. Der "Signalfluß" wird durch Pfeile angedeutet. Stellen, an denen sich Pfeile treffen, werden als Knoten bezeichnet. An jedem Knoten (bis auf Ausgangs- und Endwerte) beginnen und enden zwei Pfeile. Der Wert eines Knotens berechnet sich daher aus der Linearkombination zweier Werte der vorherigen Stufe. Die Werte von je einem horizontalen und einem von unten kommenden Pfeil werden mit Plus (gerader Anteil), die von oben kommenden und die horizontalen mit Minus (ungerader Anteil) verknüpft und mit dem zugehörigen W_j^k multipliziert. Wegen der Form, die die Pfeile in diesem Graphen bei der Berechnung zweier dualer Wertepaare (Knoten, die Werte aus einem gemeinsamen Knoten der vorherigen Stufe bekommen) bilden, wird diese Vorgehensweise des FFT-Algorithmus auch oft als Schmetterlings-/Butterfly-Operation bezeichnet.

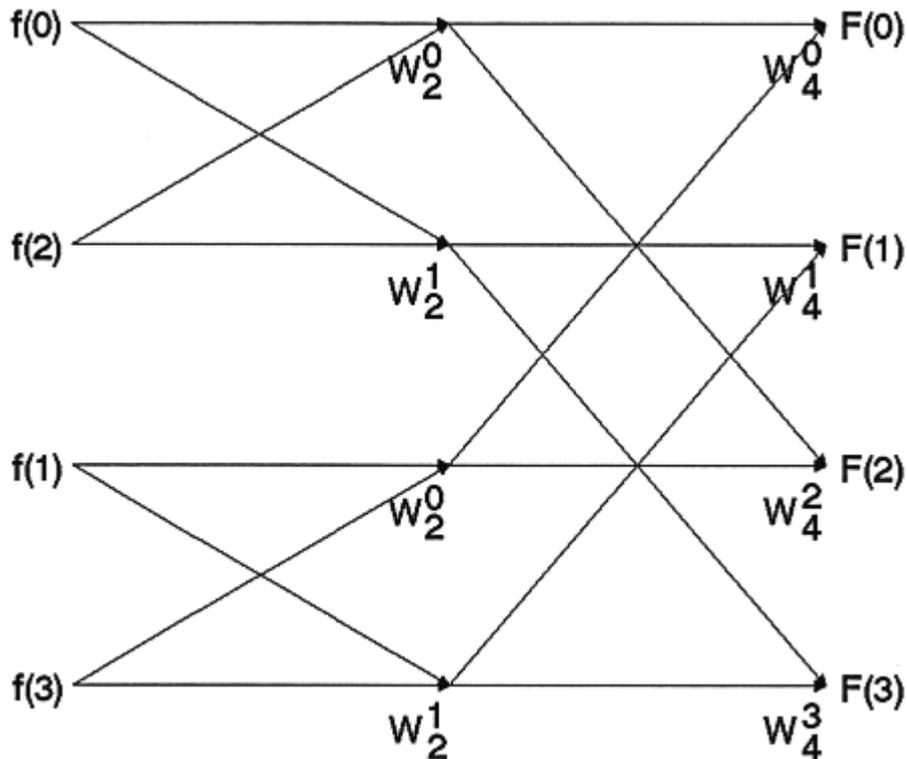


Abbildung 8.7: Der Ablauf der Verknüpfung der einzelnen Terme beim FFT-Algorithmus kann auch graphisch dargestellt werden. Zur übersichtlicheren Darstellung der Verknüpfung müssen die Eingangswerte dabei entsprechend sortiert sein (Bit-Reverse-Shuffling). Die Werte von je einem horizontalen und einem von unten kommenden Pfeil werden mit Plus, die von oben und die horizontalen mit Minus verknüpft und mit dem zugehörigen W_j^k multipliziert.

Da die Fourier-Transformation separierbar ist, geht man bei der FFT für Bilder (zweidimensionale FFT) prinzipiell so vor, daß zunächst jede Zeile des Bildes transformiert wird und die transformierten Daten noch einmal in Spalten-Richtung transformiert werden.

Im Ortsfrequenzraum können mit diesen Ergebnissen Filterungen usw. durchgeführt werden. Durch Anheben der hohen "Frequenzanteile" läßt sich zum Beispiel die Bildschärfe erhöhen. Solche Manipulationen lassen sich sogar gezielt an bestimmten Strukturen mit bestimmten Orientierungen im Ortsfrequenzraum vornehmen. Auch lassen sich unter Umständen Abbildungsfehler von Kameraobjektiven korrigieren. Nach der Manipulation der Koeffizienten erfolgt eine Rücktransformation durch eine erneute Fourier-Transformation (Vertauschungseigenschaft), um die "verbesserte" Bildinformation zurückzugewinnen.

Neben den vorgestellten Gebieten der Bildverbesserung und Bildauswertung findet die Fourier-Transformation auch Anwendung in der Bilderzeugung. So führen z.B. bei MRT-Untersuchungen (*Magnet-Resonanz-Tomographie*) die Atome des untersuchten Körpers nach einer speziellen elektromagnetischen Anregung eine Bewegung aus. Hierbei werden Signale mit bestimmter Frequenz- und Phaseninformation ausgesandt. Schließlich wird dann die übliche zweidimensionale FFT angewandt, so daß man Bilder eines Schnittes durch das Objekt erhält.

8. Fourier-Transformation

```
void FFT(int BildNr, int BildNr2)
/* BildNr = Nummer des zu transformierenden Bildes */
/* BildNr2 = Nummer des Bildes fuer das Amplitudenspektrum */
{
    int z, s; /* Zeilen-/Spalten-Index */
    int Zwerg; /* Hilfsvariable fuer Zwischenergebnisse */
    int Anzahl, ld_Anzahl;
    unsigned char BitReverse, ReverseShuffel, BitTest;
    unsigned char BitReverseShuffel[RES_MAX];
    float Faktor_C; /* Skalierungsfaktor fuer die Darstellung*/
    float Dat_Real[RES_MAX]; /* Hilfs-Vektoren fuer die Transf. */
    float Dat_Imag[RES_MAX];
    float Temp_Real[RES_MAX][RES_MAX]; /* Zwischenergebnisse d. Tr.*/
    float Temp_Imag[RES_MAX][RES_MAX];

    ClearTextWindow(35,18,80,25);
    WriteText(35,18, " FFT - zweidimensional *");

    /* Annahme: Quadratische Bilder, d.h. Zeilen == Spalten */
    Anzahl = Picture[BildNr].Zeilen;

    /* Logarithmus zur Basis 2 fuer den Faktor k (N=2^k) ermitteln */
    ld_Anzahl = 0;
    while (Anzahl > (1<<ld_Anzahl)) ld_Anzahl++;

    /* Kontrolle, ob Bild 2er-Potenz in der Aufloesung u. quadrat. */
    if ( (Anzahl != Bild[BildNr1].Spalten) ||
        (Anzahl != (1<<ld_Anzahl)) ) {
        ShowError("Falsche Aufloesung des Bildes");
        return;
    }
    Faktor_C = 10*(float)WEISS / log10((float) Anzahl + 1.0);

    /* Vektor mit den umsortierten Eintragen fuer das Bit-Reverse- */
    /* Shuffling bestimmen. */
    for (i=0; i<Anzahl; i++) {
        ReverseShuffel=0;
        BitReverse= (1<<(ld_Anzahl-1));
        BitTest =1;
        /* Jede Stelle der Binaerzahl von rechts auf 1 testen und */
        /* bei postivem Ergebnis die 1 an der entsprechenden Stelle*/
        /* von links setzen (=Bit-Reverse-Shuffling) */
        for (j=0; j<ld_Anzahl; j++) {
            if (i & BitTest) /* Bit-weise UND-Verknuepfung*/
                ReverseShuffel = ReverseShuffel | BitReverse;
            BitTest <<=1;
            BitReverse >>=1;
        }
        BitReverseShuffel[i]=ReverseShuffel;
    }
}
```

8.3 Fast- Fourier-Transformation (FFT)

```

/* Zuerst zeilenweise Fourier-Transformieren und ... */
WriteText(35,23,"Transformation in den Frequenzraum");
WriteText(35,24,"Zeile      ");
for (z=0; z<Anzahl; z++) {
    gotoxy(42,24);
    WriteInt(z);          /* aktuellen Stand anzeigen */

    /* Zur einf. Transformation Werte in Hilfs-Vektoren kopier.*/
    for (s=0; s<Anzahl; s++) {
        Dat_Real[BitReverseShuffel[s]]=Picture[BildNr].Bild[z][s];
        Dat_Imag[BitReverseShuffel[s]]=0.0;
    }
    FFT_1Dim(TRUE, Anzahl, ld_Anzahl, Dat_Real, Dat_Imag);

    /* Ergebnisse jetzt in temporaere Felder zurueckkopieren */
    for (s=0; s<Anzahl; s++) {
        Temp_Real[z][s] = Dat_Real[s];
        Temp_Imag[z][s] = Dat_Imag[s];
    }
}

/* ... das Ergebnis dann spaltenweise Fourier-Transformieren */
WriteText(35,24,"Spalte      ");
for (s=0; s<Anzahl; s++) {
    gotoxy(42,24);
    WriteInt(s);          /* aktuellen Stand anzeigen */

    /* Zur einf. Transformation Werte in Hilfs-Vektoren kopier.*/
    for (z=0; z<Anzahl; z++) {
        Dat_Real[BitReverseShuffel[z]]=Temp_Real[z][s];
        Dat_Imag[BitReverseShuffel[z]]=Temp_Imag[z][s];
    }
    FFT_1Dim(TRUE, Anzahl, ld_Anzahl, Dat_Real, Dat_Imag);

    /* Ergebnisse jetzt in temporaere Felder zurueckkopieren */
    for (z=0; z<Anzahl; z++) {
        Temp_Real[z][s] = Dat_Real[z];
        Temp_Imag[z][s] = Dat_Imag[z];
    }

    /* Amplitudenspektrum sofort mitberechnen und als Bild sp. */
    for (z=0; z<Anzahl; z++) {
        Zwerg = (int)(Faktor_C *
            log10(1.0+sqrt(Dat_Real[z]*Dat_Real[z] +
                Dat_Imag[z]*Dat_Imag[z]))) );
        /* Klipping der Grauwerte, falls diese ausserhalb lieg.*/
        if (Zwerg<SCHWARZ)    Zwerg=SCHWARZ;
        else if (Zwerg>WEISS) Zwerg=WEISS;
    }
}

```

8. Fourier-Transformation

```
/* Umordnung der Werte zur gewohnten symmetrischen */
/* Anordnung mit dem Symmetriezentrum in der Bildmitte */
if (z<Anzahl/2) {
    if (s<Anzahl/2)
        Picture[BildNr2].Bild[z+Anzahl/2][s+Anzahl/2]=Zwerg;
    else
        Picture[BildNr2].Bild[z+Anzahl/2][s-Anzahl/2]=Zwerg;
}
else {
    if (s<Anzahl/2)
        Picture[BildNr2].Bild[z-Anzahl/2][s+Anzahl/2]=Zwerg;
    else
        Picture[BildNr2].Bild[z-Anzahl/2][s-Anzahl/2]=Zwerg;
}
}
}
/* Das Amplitudenspektrum kann jetzt als Bild angezeigt werden */

/* Fuer eine Filterung muesste eine Filterform definiert werden*/
/* (Kreis, Rechteck) und die Werte, die ausserhalb dieser Form */
/* liegen auf Null gesetzt werden (Tiefpass) bzw. innerhalb auf*/
/* Null gesetzt werden (Hochpass), jeweils im Real- und Imagi- */
/* naerteil (Temp_Real, Temp_Imag). Die Ruecktransformation der*/
/* so gefilterten Daten erfolgt durch eine erneute FFT. */
}
```

```
void FFT_1Dim(int Vorwaerts_Trafo, int Anzahl, int ld_Anzahl,
             float Dat_Real[], float Dat_Imag[])
/* Vorwaerts_Trafo = Flag, ob Vorwaerts- oder Rueckwaertstransf. */
/* Anzahl          = Anzahl der Punkte (Stuetzstellen)          */
/* ld_Anzahl       = 2er-Logarithmus der Anzahl                */
/* Dat_Real[]      = Real-Anteil der Daten                      */
/* Dat_Imag[]      = Imaginaer-Anteil der Daten                */
{
    double Phi, Phi_Inkr; /* Winkel und Inkrement */
    int i,j,k,l; /* Index-Variablen */
    int m, Haelfte;
    float W_Imag, W_Real; /* Real-/Imaginaerteil der Wurzeln */
    float Tmp_Real, Tmp_Imag; /* Temporaere Variablen */

    for (i=0; i<ld_Anzahl; i++) {
        m = 2; /* gibt die Stufe der Verknuepfung an */
        m <= i;
        Haelfte = m >> 1; /* Haelfte = m/2 fuer Symmetriezwecke */
        Phi = 0.0;
        Phi_Inkr = 2.0 * PI / (double)(m);
```

8.3 Fast- Fourier-Transformation (FFT)

```
/* iteratives Vorgehen, statt rekursiv wie in Definition */
for (j=0; j<Haelfte; j++) {
    W_Real = (float)cos(Phi);
    W_Imag = (float)sin(Phi);
    if (Vorwaerts_Trafo) W_Imag = -W_Imag;
    Phi -= Phi_Inkr;
    k=j;
    do {
        l=k+Haelfte;
        Tmp_Real = W_Real*Dat_Real[l] - W_Imag*Dat_Imag[l];
        Tmp_Imag = W_Real*Dat_Imag[l] + W_Imag*Dat_Real[l];
        /* Eine Haelfte berechnen ... */
        Dat_Real[l] = Dat_Real[k] - Tmp_Real;
        Dat_Imag[l] = Dat_Imag[k] - Tmp_Imag;
        /* ... die andere Haelfte ergibt sich dann entspr. */
        Dat_Real[k] += Tmp_Real;
        Dat_Imag[k] += Tmp_Imag;
        k += m;
    } while ( k < Anzahl );
}
}
/* Normierung der Ergebnisse, je nach Definition notwendig */
if (Vorwaerts_Trafo) {
    for (j=0; j<Anzahl; j++) {
        Dat_Real[j] /=(float)Anzahl;
        Dat_Imag[j] /=(float)Anzahl;
    }
}
}
```

Aufgaben

Aufgabe 1

Wie ändert sich das Fourier-Spektrum, wenn das Objekt im Ortsraum gedreht wird?

Aufgabe 2

Gegeben sei folgende eindimensionale Rechteckfunktion:

$$f(x) = \begin{cases} A & 0 < x \leq X \\ 0 & \text{sonst} \end{cases}$$

Berechnen Sie die Fourier-Transformierte und das Fourier-Spektrum dieser Funktion.

9. Bildtransformationen

Bildtransformationen sind nützlich, um bestimmte Eigenschaften eines Bildes besser hervorzuheben. So wurde z.B. schon bei der Fourier-Transformation deutlich, daß damit einfach Resonanzfrequenzen oder dominierende Frequenzen einer Funktion festgestellt, bestimmte Frequenzen verstärkt, gedämpft oder ganz unterdrückt werden können.

Die nachfolgend besprochenen Transformationen beruhen alle auf demselben Prinzip, weisen jedoch zum Teil einige Vereinfachungen auf (keine komplexen Zahlen, in einigen Fällen nur Ganzzahl-Arithmetik), um mit möglichst geringen qualitativen Verlusten und gleichzeitig möglichst geringem rechnerischen Aufwand diese Transformationen durchzuführen. Damit diese Transformationen überhaupt anwendbar sind, muß die Rücktransformation möglich sein und die Inverse des Vorwärtstransformationskernes (siehe unten) existieren und leicht zu bestimmen sein. Anders ausgedrückt bedeutet dies, daß die Basis, die durch diese Transformationen gebildet wird, den ganzen Vektorraum aufspannen muß. Dies ist jedoch nur möglich, falls alle M Basisvektoren senkrecht aufeinander stehen, d.h. eine Orthogonalbasis bilden.

$$K K^{T*} = M I$$

Eine Matrix K , deren Inverse K^{-1} ihrer konjugiert Transponierten K^{T*} gleich, wird als unitäre Matrix bezeichnet. Aus diesem Grund nennt man diese Art der Transformationen auch *Unitäre Transformationen*.

Dabei können einige der zweidimensionalen Transformationen in der Form

$$T(k, l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) g(m, n, k, l) \quad m, k = 0, 1, \dots, M-1 \\ n, l = 0, 1, \dots, N-1$$

geschrieben werden, analog die Rücktransformation. $T(k, l)$ ist die Transformierte der Funktion $f(m, n)$ und $g(m, n, k, l)$ der sogenannte Vorwärtstransformationskern.

9.1 Diskrete Fourier-Transformation

Die *Fourier-Transformation* wurde im vorigen Abschnitt schon ausführlich vorgestellt. Sie ermöglicht es mit Matrizen im Frequenzbereich statt mit Integralen im Ortsbereich zu rechnen. Bei teilweiser Kenntnis der Originalfunktion (Originalbild) ist es möglich, Störungen dieser Funktion durch mathematische Operationen, nämlich durch Modifikation des Frequenzspektrums, weitgehend zu beheben.

Die eindimensionale DFT lautet

9. Bildtransformation

$$F(k) = \frac{1}{M} \sum_{m=0}^{M-1} f(m) e^{(-i2\pi km)/M} \quad m, k = 0, 1, \dots, M-1$$

Dabei entspricht der Term $e^{(-i2\pi km)/M}$ dem Vorwärtstransformationskern. Über die Euler-Identität läßt sich dieser Ausdruck in Sinus- und Kosinusanteile zerlegen. Diese Wellen mit verschiedenen Wellenlängen stellen die sogenannten Basisfunktionen dar. Die Zahl k (Wellenzahlindex) gibt an, wie oft die Periode der Basisfunktion in das Intervall $[0, M-1]$ hineinpaßt. Die Basisfunktionen für $M = 16$ sind in Abbildung 9.1 gezeigt.

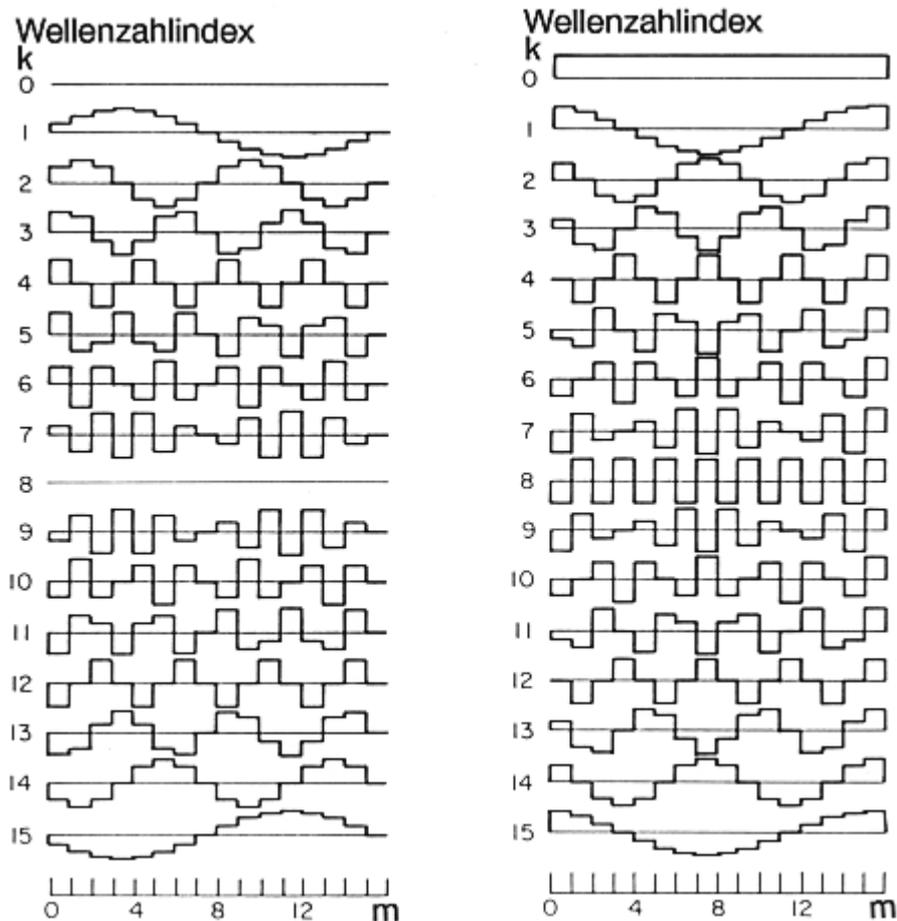


Abbildung 9.1: Die Basisfunktionen der eindimensionalen diskreten Fourier-Transformation (DFT), zerlegt in den Realteil (Kosinus; links) und den Imaginärteil (Sinus; rechts), dargestellt für $M = 16$.

Für die niedrigen Frequenzen stellen diese Basisfunktionen eine grobe Annäherung der Sinusfunktion dar. Bei höheren Frequenzen gehen sie langsam in Rechteckfunktionen über.

Wie schon bei der FFT benutzt, wird auch hier im Transformationskern die Abkürzung

$$W_M = e^{(-i2\pi)/M}$$

verwendet. In der Regel ist M eine Zweierpotenz. Es gilt dann $W_M^M = 1$, $W_M^{M/2} = -1$, $W_M^{M/4} = -i$ und $W_M^{3M/4} = i$. In der Matrixschreibweise kann der Transformationskern wie folgt abgekürzt werden

$$W = \frac{1}{M} \begin{bmatrix} W^0 & W^0 & W^0 & \dots & W^0 \\ W^0 & W^1 & W^2 & \dots & W^{M-1} \\ W^0 & W^2 & W^4 & \dots & W^{2(M-1)} \\ \dots & \dots & \dots & \dots & \dots \\ W^0 & W^{M-1} & W^{2(M-1)} & \dots & W^{(M-1)(M-1)} \end{bmatrix}$$

Zur besseren Übersichtlichkeit ist der Index M an den einzelnen W^k weggelassen worden. In dieser Matrixschreibweise lautet die DFT

$$F = Wf$$

Die zweidimensionale DFT ist analog zur eindimensionalen gegeben durch

$$F(k, l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-i2\pi km/M} e^{-i2\pi ln/N} \quad \begin{array}{l} m, k = 0, 1, \dots, M-1 \\ n, l = 0, 1, \dots, N-1 \end{array}$$

$$F(k, l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) W_M^{km} W_M^{ln} \quad \begin{array}{l} m, k = 0, 1, \dots, M-1 \\ n, l = 0, 1, \dots, N-1 \end{array}$$

Da die Transponierte des Transformationskerns W identisch mit W ist, lautet die zweidimensionale DFT aufgrund der Separierbarkeit

$$F = WfW^T$$

Die Rücktransformation erfolgt jeweils mit der konjugierten transformierten Matrix.

9.2 Diskrete Sinus/Kosinus-Transformation

Durch die diskrete Fourier-Transformation werden die Ergebnisse komplex, da bei der Transformation nur der Kosinus-Term einen reellen Anteil liefert. Durch die Zusammensetzung der Transformation aus einem reellen geraden (cos) und einem imaginären ungeraden (sin) Teil wird dies noch einmal verdeutlicht.

$$W_M^{km} = \cos\left(-\frac{2\pi km}{M}\right) + i \sin\left(-\frac{2\pi km}{M}\right)$$

Keine der beiden Teiltransformationen ist umkehrbar, da im Realteil nur die geraden und im Imaginärteil nur die ungeraden Funktionsanteile zu finden sind.

Ein Trick eröffnet hier aber einen Ausweg. Die Funktion wird künstlich gerade oder ungerade gemacht, so daß eine Teiltransformation ausreicht. Man stellt sich dazu das Bild künstlich auf die doppelte Länge/Breite vergrößert vor. Dadurch werden dann die Kerne der beiden Teiltransformationen reell.

9. Bildtransformation

Diese Vergrößerung geschieht entweder durch Spiegelung des Ursprungsbildes am Rand, wodurch aus einem $N \times N$ -Bild ein $2N \times 2N$ -Bild wird oder durch Spiegelung an der Randpixelzeile. Diese Randpixelzeile wird dann von allen Teilen, Originalbild und gespiegeltem Bild gemeinsam genutzt. Es entsteht so ein vergrößertes Bild der Größe $(2N-1) \times (2N-1)$. Entsprechend dieser künstlichen Vergrößerung wird die danach durchgeführte Transformation als *gerade* oder *ungerade* bezeichnet. Man unterscheidet daher zwischen der geraden und ungeraden *diskreten Sinus-Transformation* (DST) bzw. *diskreten Kosinus-Transformation* (DCT). Diese lauten:

ungerade Kosinus-Transformation:

$$C(k, l) = \frac{2}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \cos\left(\frac{\pi}{N} k \left(m + \frac{1}{2}\right)\right) \cos\left(\frac{\pi}{N} l \left(n + \frac{1}{2}\right)\right)$$

gerade Kosinus-Transformation:

$$C(k, l) = \frac{4}{2N-1} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \cos\left(\frac{2\pi}{2N-1} km\right) \cos\left(\frac{2\pi}{2N-1} ln\right)$$

ungerade Sinus-Transformation:

$$S(k, l) = \frac{2}{N+1} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \sin\left(\frac{(m+1)(k+1)\pi}{N+1}\right) \sin\left(\frac{(n+1)(l+1)\pi}{N+1}\right)$$

gerade Sinus-Transformation:

$$S(k, l) = \frac{1}{N+1} \sum_{m=0}^{2N+1} \sum_{n=0}^{2N+1} f(m, n) \sin\left(\frac{2(k+1)m}{2N+2}\right) \sin\left(\frac{2(n+1)l}{2N+2}\right)$$

Da sowohl die Basisfunktionen der Kosinus-Transformation als auch der Sinus-Transformation separierbar sind, kann die zweidimensionale Transformation analog zur Fourier-Transformation durch zweimalige Anwendung der eindimensionalen Transformation nach Zeilen und Spalten berechnet werden.

9.3 Walsh-Transformation

Die Walsh-Transformation ordnet dem untersuchten Signal die Amplituden der einzelnen sogenannten Walsh-Schwingungen zu, die dieses Signal enthält. Für die diskrete Walsh-Transformation und die endliche Walsh-Transformation gilt das Gleiche wie für die entsprechenden Fourier-Transformationen: Im diskreten Fall wird ein endlicher, periodischer Prozeß gut approximiert.

Die zweidimensionale Walsh-Transformation ist mit $M = 2^p$ und $N = 2^q$ definiert als

$$W(k, l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \prod_{i=0}^{p-1} (-1)^{b_i(m)b_{p-1-i}(k)} \prod_{i=0}^{q-1} (-1)^{b_i(m)b_{q-1-i}(l)}$$

Dabei ist $b_i(m)$ das i -te Bit der binären Zahlendarstellung von m . Dadurch enthält der Vorwärtstransformationskern nur die Werte $+1$ und -1 . In der Matrixschreibweise gibt man dann häufig nur noch das Vorzeichen an. Prinzipiell läßt sich die Walsh-Transformation aber für beliebige $M, N > 0$ definieren.

Im eindimensionalen Fall hat der Vorwärtstransformationskern für $M = 8$ folgendes Aussehen:

$$W = \frac{1}{M} \begin{bmatrix} + & + & + & + & + & + & + & ++ \\ + & + & + & + & - & - & - & - \\ + & + & - & - & + & + & - & - \\ + & + & - & - & - & - & + & + \\ + & - & + & - & + & - & + & - \\ + & - & + & - & - & + & - & + \\ + & - & - & + & + & - & - & + \\ + & - & - & + & - & + & + & - \end{bmatrix}$$

Da der Kern der Walsh-Transformation symmetrisch ist und die Zeilen und Spalten orthogonal sind (Orthogonalbasis), ist der Rückwärtstransformationskern bis auf den Faktor $1/M$ identisch mit diesem Vorwärtstransformationskern.

Analog zur Fast-Fourier-Transformation wird die Fast-Walsh-Transformation (FWT) definiert.

$$\begin{aligned} W(u) &= \frac{1}{2} \{W_g(u) + W_u(u)\} \\ W(u+N) &= \frac{1}{2} \{W_g(u) - W_u(u)\} \end{aligned}$$

mit $N = M/2$, $u = 0, 1, \dots, M-1$. Somit läßt sich ein analoges Transformationsnetzwerk für die FWT wie für die FTT konstruieren.

Die Güte der Approximation durch Walsh-Funktionen reicht für bestimmte Zwecke z.B. in der Bildverarbeitung oft aus, so daß wegen der leichten Handhabung und der gegenüber der Fourier-Transformation viel schnelleren und weitaus weniger aufwendigen Berechnung diese Approximationsart häufig angewandt wird [GW87].

9.4 Hadamard-Transformation

Die *Hadamard-Transformation* ist der Walsh-Transformation sehr ähnlich. Auch ihr Transformationskern besteht nur aus den Werten $+1$ und -1 . Sie ist definiert als

9. Bildtransformation

$$H(k, l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) (-1)^{\sum_{i=0}^{p-1} b_i(m) b_i(k)} (-1)^{\sum_{i=0}^{q-1} b_i(n) b_i(l)}$$

Hierbei ist auch $b_i(m)$ das i -te Bit der binären Zahlendarstellung von m . Dadurch enthält dieser Vorwärtstransformationskern ebenfalls nur die Werte $+1$ und -1 .

Der große Vorteil der Hadamard-Transformation ist, daß bei quadratischen Ausgangsfunktionen mit $N = M$ und $N = 2^p$ der Transformationskern rekursiv nach der Formel:

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix}$$

berechnet werden kann.

Die Walsh- und Hadamard-Transformation zerlegen die Originalfunktion nicht in Sinus- und Kosinusanteile, sondern nähern diese durch Rechteckfunktionen an.

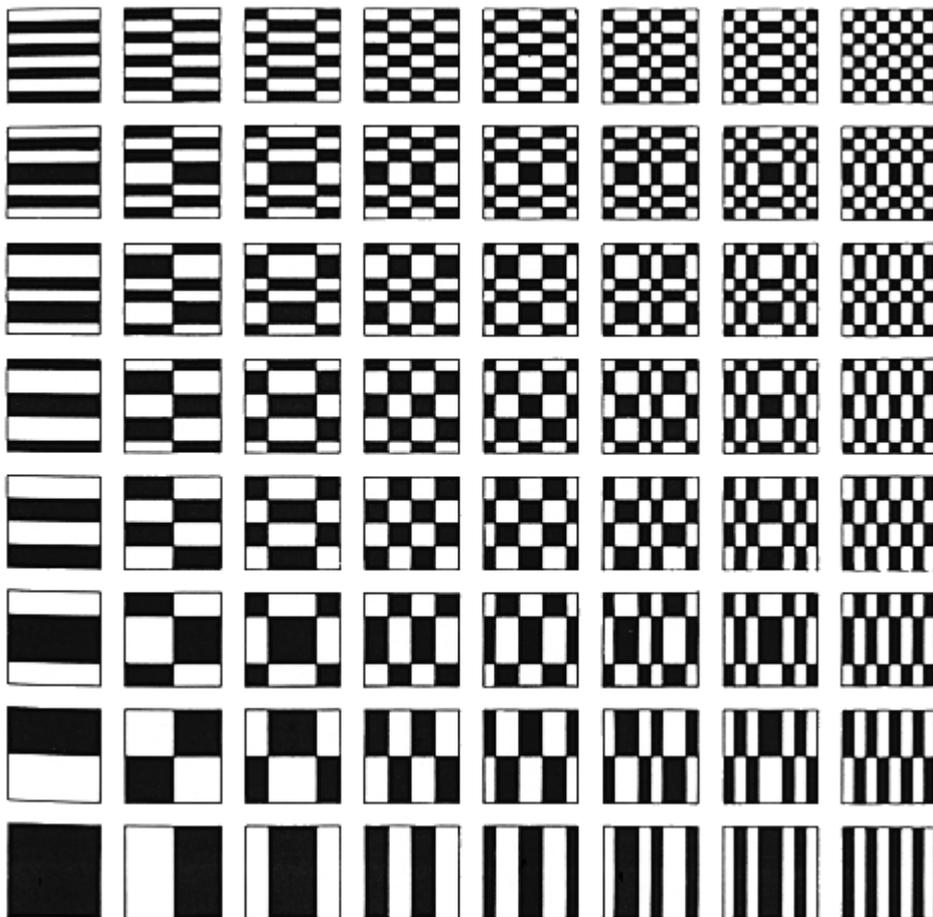


Abbildung 9.2: Darstellung des Transformationskerns der zweidimensionalen Hadamard-Transformation für $N = 8$ als Schwarzweißbild. Schwarz= $+1$; Weiß= -1 .

```

void Hadamard(int BildNr, int BildNr2)
/* BildNr = Nummer des zu transformierenden Bildes */
/* BildNr2 = Nummer des Bildes fuer das Amplitudenspektrum */
{
int z, s; /* Zeilen-/Spalten-Index */
int Zwerg, n; /* Hilfsvariable fuer Zwischenergebnisse */
int Anzahl, s, z; /* Anzahl Zeilen/Spalten des Bildes */
float Faktor_C; /* Skalierungsfaktor fuer die Darstellung*/
long Temp[RES_MAX][RES_MAX]; /* Zwischenergebnisse d. Transf. */
long Temp1[RES_MAX][RES_MAX];
long Hadam[RES_MAX][RES_MAX]; /* Hadamard-Matrix mit +1 und -1 */

ClearTextWindow(35,18,80,25);
WriteText(35,18,"* Hadamard - zweidimensional *");

/* Annahme: Quadratische Bilder, d.h. Zeilen == Spalten */
Anzahl = Picture[BildNr].Zeilen;

/* Logarithmus zur Basis 2 fuer den Faktor k (N=2^k) ermitteln */
ld_Anzahl = 0;
while (Anzahl > (1<<ld_Anzahl)) ld_Anzahl++;

/* Kontrolle, ob Bild 2er-Potenz in der Aufloesung u. quadrat. */
if ( (Anzahl != Bild[BildNr1].Spalten) ||
      (Anzahl != (1<<ld_Anzahl)) ) {
    ShowError("Falsche Aufloesung des Bildes");
    return;
}
Faktor_C = 10*(float)WEISS / log10((float) Anzahl + 1.0);

/* Initialisierung der Hadamard-Matrix. Diese ist rekursiv */
Hadam[0][0] = 1; /* definiert mit der Initialisierung: 1 1 */
Hadam[0][1] = 1; /* (Ordnung 2) 1 -1 */
Hadam[1][0] = 1;
Hadam[1][1] = -1;
/* Erzeugung der Hadamard-Matrix in der richtigen Groesse */
n=2;
while (n<=(Anzahl/2)) {
    for (z=0; z<n; z++) /* Teil rechts oben */
        for (s=n; s<2*n; s++) Hadam[z][s] = Hadam[z][s-n];
    for (z=n; z<2*n; z++) /* Teil links unten */
        for (s=0; s<n; s++) Hadam[z][s] = Hadam[z-n][s];
    for (z=n; z<2*n; z++) /* Teil rechts unten */
        for (s=n; s<2*n; s++) Hadam[z][s] = -Hadam[z-n][s-n];
    n *= 2;
}
for (z=0; z<Picture[BildNr].Zeilen; z++) /* Originalbild kopieren */
    for (s=0; s<Picture[BildNr].Spalten; s++) {
        Temp[z][s] = Picture[BildNr].Bild[z][s];

/* Hadamard-Transformation: Temp = Hadam * Bild * Hadam */
/* Zuerst Zeilen transformieren: Temp1 = Hadam * Bild */
MatrixMultiplikation( Hadam, Temp, Temp1, Anzahl );

```

9. Bildtransformation

```
/* Dann Spalten transformieren: Temp = Temp1 * Hadam */
MatrixMultiplikation( Temp1, Hadam, Temp, Anzahl);

/* Amplitudenspektrum mitberechnen und als Bild speichern */
for (z=0; z<Anzahl; z++) {
    for (s=0; s<Anzahl; s++) {
        /* Nur Phaseninformation vorhanden, keine Amplitude */
        Zwerg = fabs( (float)Temp[z][s] / (float)Anzahl );
        Zwerg = (int)(Faktor_C * log10(1.0 + Zwerg));

        /* Klipping der Grauwerte, falls diese ausserhalb lieg.*/
        if (Zwerg<SCHWARZ) Zwerg=SCHWARZ;
        else if (Zwerg>WEISS) Zwerg=WEISS;

        Picture[BildNr2].Bild[z][s] = Zwerg;
    }
}
/* Das Amplitudenspektrum kann jetzt als Bild angezeigt werden */
}
```

9.5 Hauptachsen-Transformation

Die Hauptachsen-Transformation (auch Karhunen-Loève-, Hotelling -, Prinzipal- oder Eigenvektor-Transformation genannt) wird in der Hauptsache zur Datenkompression, zur Bilddrehung (standardisierte Darstellung) und zur Klassifizierung angewandt. Dafür müssen Merkmale benutzt werden, die möglichst miteinander unkorreliert sind. Diese Art der Transformation eines kontinuierlichen Signals wurde ursprünglich von Karhunen und von Loève entwickelt. Hotelling hat sich mit der Transformation eines diskreten Signals in einen Satz unkorrelierter Koeffizienten beschäftigt.

Zur Berechnung werden ausgehend von einem "Norm-" oder "Mittelwertbild" die Kovarianzmatrix und die dazugehörigen Eigenwerte und Eigenvektoren berechnet. Die Eigenvektoren repräsentieren die einzelnen Merkmale der Bildklasse und bilden eine Matrix, die zusammen mit dem Mittelwertbild die Transformation auf das zu transformierende Bild ergeben. Das Ergebnis dieser Hauptachsen-Transformation ist dann eine Matrix, die Gewichtungsfaktoren für die Eigenvektoren enthält. Über diese Gewichtungsfaktoren können Aussagen über das Vorkommen und die Wichtigkeit bestimmter Merkmale im zu untersuchenden Bild gemacht werden. Je größer der Faktor, desto wichtiger ist das korrespondierende Merkmal. Oft reichen daher einige wenige dieser Faktoren aus, um ein Bild relativ genau zu beschreiben. So werden z.B. durch die sechs größten Eigenwerte zusammen mit dem Mittelwertbild 97% eines Bildes bei Satellitenbildern (LANDSAT) von der Erde beschrieben.

Bei dieser Transformation faßt man im einzelnen das Bildsignal als ein $M N$ -dimensionales Zufallssignal mit entsprechend $M N$ verschiedenen Merkmalen auf. Durch eine Stichprobe $Q = \{f_1(x, y), \dots, f_M(x, y)\}$ von diesen Bildern wird das Norm- oder Mittelwertbild μ bestimmt. Ohne

Einschränkung der Allgemeingültigkeit der nachfolgenden Berechnungen soll $M = N$ angenommen werden, es werden also nur quadratische Bilder betrachtet. Dies vereinfacht an einigen Stellen die Schreibweise. Der "Vektor" \vec{f} hat dann die Dimension N^2 und repräsentiert ein zufälliges Bildsignal, entweder zeilen- oder spaltenweise.

$$\bar{\mu}_f \cong \frac{1}{M} \sum_{i=1}^M \vec{f}_i = E(f)$$

Über diese Stichprobe und dem angenäherten Mittelwert wird die Kovarianzmatrix C_f abgeschätzt.

$$\begin{aligned} C_f &\cong E\{(\vec{f} - \bar{\mu}_f)(\vec{f} - \bar{\mu}_f)^T\} \\ &\cong \frac{1}{M} \sum_{i=1}^M (\vec{f}_i - \bar{\mu}_f)(\vec{f}_i - \bar{\mu}_f)^T \\ &\cong \frac{1}{M} \left(\sum_{i=1}^M \vec{f}_i \vec{f}_i^T \right) - \bar{\mu}_f \bar{\mu}_f^T \end{aligned}$$

Da sowohl \vec{f} als auch $\bar{\mu}_f$ von der Dimension N^2 sind, ist diese Kovarianzmatrix von der Dimension $N^2 \times N^2$.

Nun sind die Eigenvektoren \vec{e}_i und die Eigenwerte λ_i zu berechnen, so daß gilt

$$\begin{aligned} C_f \vec{e}_i &= \lambda_i \vec{e}_i \\ (C_f - \lambda_i I) \vec{e}_i &= 0 \end{aligned}$$

Ist die Determinante $\det(C_f - \lambda_i I) = 0$, so sind die Zeilen der Kovarianzmatrix linear abhängig und es läßt sich keine eindeutige Lösung berechnen.

Die Eigenwerte λ_i dieser Kovarianzmatrix werden der Größe nach geordnet $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{N^2}$ und die Elemente der Eigenvektoren \vec{e}_i in eine Matrix A eingetragen. Diese in den Spalten von A stehenden Eigenvektoren nennt man auch die Hauptachsen und entsprechend diese Transformation die Hauptachsen-Transformation.

Über die Hotelling-Transformation

$$\vec{g} = A(\vec{f} - \bar{\mu}_f)$$

erhält man aus dem mittelwertfreien Bildvektor durch Multiplikation mit A einen neuen Bildvektor, der die Koeffizienten enthält, wie die einzelnen Eigenvektoren zu gewichten sind.

$$A = \begin{bmatrix} e_{11} & e_{12} & e_{13} & \dots & e_{1N^2} \\ e_{21} & e_{22} & e_{23} & \dots & e_{2N^2} \\ e_{31} & e_{32} & e_{33} & \dots & e_{3N^2} \\ \dots & \dots & \dots & \dots & \dots \\ e_{N^21} & e_{N^22} & e_{N^23} & \dots & e_{N^2N^2} \end{bmatrix}$$

9. Bildtransformation

Dabei bedeutet e_{ij} die i -te Komponenten des j -ten Eigenvektors. Da die Kovarianzmatrix C_f eine reelle, symmetrische Matrix ist, läßt sich immer ein Satz reeller orthonormaler Eigenvektoren finden, so daß $A^{-1} = A^T$ gilt.

Durch Einsetzen in die obigen Gleichungen kann man zeigen, daß der Erwartungswert $\bar{\mu}_g = E(g)$ gleich Null ist. Somit ergibt sich für C_g

$$\begin{aligned} C_g &= E\{(\bar{g} - \bar{\mu}_g)(\bar{g} - \bar{\mu}_g)^T\} \\ &= E\{(A\bar{f} - A\bar{\mu}_f)(A\bar{f} - A\bar{\mu}_f)^T\} \\ &= E\{(A\bar{f} - A\bar{\mu}_f)(\bar{f} - \bar{\mu}_f)^T A^T\} \\ &= A E\{(\bar{f} - \bar{\mu}_f)(\bar{f} - \bar{\mu}_f)^T A^T\} \\ &= A C_f A^T \end{aligned}$$

Die Kovarianzmatrix C_g von der Dimension $N^2 \times N^2$ ist eine Diagonalmatrix mit den Elementen

$$c_{ij} = \begin{cases} \lambda_i & i = j \\ 0 & \text{sonst} \end{cases}$$

Diese Eigenwerte entsprechen der Varianz der Merkmale, wobei der erste Eigenwert λ_1 die größte Varianz aufweist und somit das aussagekräftigste Merkmal repräsentiert. Da die anderen Werte (Kovarianzen) von C_g Null sind, bedeutet dies, daß die Merkmale unkorreliert sind. Somit läßt sich insgesamt die Rücktransformation als

$$\bar{f} = A^T \bar{g} + \bar{\mu}_f$$

definieren. Beschränkt man sich bei dieser Rücktransformation auf die ersten P Elemente von \bar{g} , so kann \bar{f} durch \bar{g}_P im Sinne der minimalen mittleren Fehlerquadrate optimal angenähert werden. Dies ist eine wichtige Eigenschaft zur Datenreduktion.

$$\begin{aligned} \mathcal{E} &= \sum_{i=1}^{N^2} \lambda_i - \sum_{i=1}^P \lambda_i = \sum_{i=P+1}^{N^2} \lambda_i \\ &= \text{Minimale Summe der Fehlerquadrate} \end{aligned}$$

Die Basis-Vektoren (z.B. Zeilen von A) sind so gewählt, daß sie in die Richtung der größten Varianz zeigen.

Somit ist es möglich, eine Transformation von dem einen in das andere Koordinatensystem zu definieren, die Objekte in einer standardisierten Darstellung wiedergibt.

$$\begin{aligned} y_1 &= x_1 \cos \alpha + x_2 \sin \alpha \\ y_2 &= -x_1 \sin \alpha + x_2 \cos \alpha \end{aligned}$$

wobei x_1, x_2 die Koordinaten (Indizes) im alten und y_1, y_2 die Koordinaten (Indizes) im neuen Koordinatensystem sind. Die Transformation

$$g = Af$$

entspricht dann einer Rotation, wenn A die zugehörige Rotationsmatrix darstellt (siehe Kapitel 4). Über die Formel

$$g = A(f - \mu_f)$$

wird zusätzlich noch eine Translation definiert. Jeder Vektor wird aus den Koordinaten der Pixel eines einzelnen Bildes gebildet. Alle Vektoren stammen von einem Objekt des Bildes.

Damit kann ein Objekt mit den Hauptausdehnungsrichtungen parallel zu den Koordinatenachsen gedreht werden. Man erhält eine normierte Lage.

Die Eigenschaft der Hotelling-Transformation, Signale in einen Raum mit orthogonalen Eigenvektoren als Basisvektoren abzubilden, wobei diese jeweils in Richtung der größten Signalvarianz zeigen, läßt sich vorteilhaft für eine standardisierte Darstellung von Objekten in Bildern ausnutzen.

In [KS90] wurde die Hauptachsen-Transformation zur Charakterisierung von Gesichtern verwendet. Aus einer genügend großen Stichprobe wurde dazu das "Normgesicht" berechnet. Mit diesem Mittelwertgesicht wurden bei einem neuen Gesicht die Eigenvektoren und Eigenwerte berechnet. Es stellte sich dabei heraus, daß etwa 50 Werte für die Charakterisierung eines Gesichtes zusammen mit dem Mittelwertgesicht ausreichen. Probleme bereiteten bei diesem Ansatz die Haare, weshalb diese meist nicht berücksichtigt wurden. Für den praktischen Einsatz bedeutet dies insgesamt, daß ein Gesicht z.B. beim Bildtelefon komplett mit den 50 Eigenwerten im Aussehen übertragen werden kann, wenn beim Empfänger das Mittelwertgesicht bekannt ist. Damit ist die Übertragung selbst komplexer Bildinhalte über langsame Medien (Telefon) möglich.

9.6 Weitere Bildtransformationen

Neben den hier genannten Transformationen finden noch andere, wie die Haar-, Slant-, SVD- (singular-valued-decomposition) Transformation usw. bei der Bildtransformation Verwendung [AH77], [Pra789]. Sie verwenden fast alle das gleiche Prinzip und benutzen nur unterschiedliche Transformationskerne.

Aufgaben

Aufgabe 1

1. Wie lautet die Hadamard-Matrix der Größe 4?
2. Stellen Sie die Transformationsgleichungen für die Hadamard-Transformation einer Folge mit 4 Elementen $f(i)$, $i=0, 1, 2, 3$ auf. Wie groß ist die Anzahl der Additionen?
3. Fassen Sie je zwei Elemente der Folge $f(i)$ so zusammen, daß eine zweistufige Berechnung der Hadamard-Transformation erfolgt und dadurch Additionen eingespart werden. Wieviel Additionen sind nun noch notwendig?
4. Zeichnen Sie das Verknüpfungs-Netzwerk dieser zweistufigen Berechnung.